



**Technische Universität Ilmenau**

Fakultät für Informatik und Automatisierung

Fachgebiet Neuroinformatik und Kognitive Robotik

# **Personendetektion durch Klassifikation von Laserscans**

Masterarbeit zur Erlangung des akademischen Grades Master of Science

**Tim Wengefeld**

Betreuer: Dipl.-Inf. Michael Volkhardt

Dipl.-Inf. Christoph Weinrich

Verantwortlicher Hochschullehrer:

Prof. Dr. H.-M. Groß, FG Neuroinformatik und Kognitive Robotik

Die Masterarbeit wurde am 21.02.2014 bei der Fakultät für Informatik  
und Automatisierung der Technischen Universität Ilmenau eingereicht.

[urn:nbn:de:gbv:ilm1-2015200035](https://nbn-resolving.org/urn:nbn:de:gbv:ilm1-2015200035)



Erklärung: „Hiermit versichere ich, dass ich diese Masterarbeit selbstständig verfasst und nur die angegebenen Quellen und Hilfsmittel verwendet habe. Alle von mir aus anderen Veröffentlichungen übernommenen Passagen sind als solche gekennzeichnet.“

Ilmenau, 21.02.2014

.....  
Tim Wengefeld





# Zusammenfassung

Die Möglichkeit eines Roboters zu erkennen ob und wo sich Menschen in seiner näheren Umgebung befinden, ist eine der wichtigsten Eigenschaften für eine möglichst natürliche Mensch-Maschine-Interaktion. Diese Fertigkeit ermöglicht es ihm erst eine auf den Menschen abgestimmte Planung seiner nächsten Handlungen vorzunehmen.

In der Vergangenheit wurden zu diesem Thema verschiedenste Verfahren entwickelt, welche unterschiedliche Sensoren Verwenden. Jedes dieser Verfahren unterliegt jedoch sensorspezifischen Einschränkungen, hinsichtlich der Umgebungsbedingungen und/oder der Berechnungskomplexität. Deshalb werden in der Praxis meist mehrere Verfahren die unterschiedliche Sensoren verwenden miteinander kombiniert um so eine möglichst für alle Situationen robuste Detektion zu gewährleisten.

Ein weit verbreiter Ansatz hierbei ist, das Erkennen spezifischer Menschlicher Merkmale in einzelnen 2D Lasermessungen, insbesondere von Beinen bei tief liegenden Lasersensoren. Ein in der Fachliteratur oft beschriebenes Problem dieser Methode ist, dass Beine von Gegenständen der Umgebung ganz oder teilweise verdeckt werden können, wofür bisher keine Lösung gefunden werden konnte. Dieses Hindernis wird ungleich Problematischer, wenn diese Gegenstände nicht Teil der Umgebungen sind, sondern Bestandteil der Erscheinungsform einer Person. So kommen in Krankenhäusern oder Altenheimen oft Laufhilfen wie Krücken oder Rollatoren zum Einsatz, die aktiv mit der Person mitgeführt werden und somit weitaus häufiger für Verdeckungen sorgen. Dieses Problem führt soweit, das Personen die in Rollstühlen sitzen eine komplett andere Erscheinungsform besitzen und nicht mehr durch ihre Beine erkannt werden können.

Diese Arbeit beschäftigt sich einerseits mit der Verbesserung der am Fachgebiet vorhandenen Implementierung des State of the Art Detektors. Außerdem wird ein neues Verfahren vorgestellt, dass das Problem der Verdeckungen behandelt und somit auch in Situationen Ergebnisse liefern kann, die von der Fachliteratur so noch nicht betrachtet wurden.

Die Arbeit schließt mit einer ausführlichen vergleichenden Evaluation, sowie den Ergebnissen für Personenhypothesen die nicht nur auf Beine beschränkt sind.

# Abstract

The ability of a robot to perceive if and where a person is located in his environment, is one of the most crucial skills to guarantee a natural Human-Robot-Interaction. This capability enables the robot to plan his actions according to a human opposit.

In the past the scientific community introduced several approaches which use different sensors to handle this problem. However, no method is able to solve this problem perfectly, because every one suffers from different drawbacks according to specific environmental conditions and/or computational complexity. In practice several methods and sensors have to be combined to ensure a robust detection.

A very common approach, for a part of those detection systems, is to recognize the specific features of a person, especially legs in lower located sensors, in a single 2D Laserscan. One often mentioned drawback of this method is that legs can often (completely or partly) be occluded by environmental objects, an issue that has not been treated so far. This problem gets highly significant if these objects are not just part of the environment but part of the appearance of the person itself. To consider operational environments like hospitals or retirement homes, where persons often take use of walking aids like crutches or wheeled walkers. This objects are carried by the person and therefore cause occlusions more often. To picture this issue more generally, a person in a wheelchair has a total different appearance in a Laserscan and cannot be identified by its legs.

One part of this thesis deals with improving the robustness of the current implementation of the State of the Art leg detector, which is already used by the department. Furthermore, an approach will be introduced which treats the problem of occlusions and therefore is able to give a more general person hypothesis.

Finally the thesis concludes with a comparative evaluation and the results for hypotheses that are not just limited to legs.



# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Motivation . . . . .	1
1.2	Problemstellung . . . . .	2
1.3	Bedeutung für Projekte am Fachgebiet . . . . .	2
1.4	Zielsetzung . . . . .	4
1.5	Überblick . . . . .	5
<b>2</b>	<b>State of the Art</b>	<b>7</b>
2.1	Sensorik für Personendetektion auf mobilen Robotersystemen . . . . .	8
2.2	Personendetektion in Laserscans . . . . .	10
2.2.1	2D Verfahren . . . . .	10
2.2.2	3D Verfahren . . . . .	15
<b>3</b>	<b>Grundlagen</b>	<b>21</b>
3.1	Grundlegende Probleme der Personendetektion . . . . .	21
3.2	Ablauf der Personendetektion . . . . .	24
3.2.1	Vorverarbeitung . . . . .	24
3.2.2	Segmentierung . . . . .	25
3.2.3	Merkmalsgenerierung . . . . .	28
3.2.4	Klassifikation . . . . .	31
3.2.5	Generierung der Personenhypothese . . . . .	35
3.2.6	Generierung der Trainingsdaten . . . . .	36
3.3	Crossvalidation und Learning Curves . . . . .	37

---

<b>4</b>	<b>Eigener Ansatz</b>	<b>43</b>
4.1	Die Grundidee . . . . .	45
4.2	Ablauf der Personendetektion . . . . .	50
<b>5</b>	<b>Evaluation</b>	<b>53</b>
5.1	Verwendete Lasersensoren . . . . .	53
5.2	Datenbasis . . . . .	53
5.3	Evaluierungsmethodik . . . . .	56
5.4	Vergleich zu Arras et. al. . . . .	60
5.5	Personenhypothesen für Beine . . . . .	63
5.6	Learning Curve Analyse . . . . .	67
5.7	Laufzeit . . . . .	69
5.8	Hypothesen für Rollstühle und Rollatoren . . . . .	70
<b>6</b>	<b>Zusammenfassung und Ausblick</b>	<b>73</b>
6.1	Zusammenfassung . . . . .	73
6.2	Ausblick . . . . .	74
<b>A</b>	<b>weitere Ergebnisse</b>	<b>77</b>
A.1	Vergleich der Adaboost Varianten . . . . .	77
A.2	Vergleich der Hypothesengenerierung . . . . .	77
A.3	Ergebnisse für den Roboter SCITOS G3 . . . . .	77
<b>B</b>	<b>Berechnungsmethoden für die lokalen Tiefenwerte</b>	<b>81</b>
<b>C</b>	<b>weitere Algorithmen</b>	<b>85</b>
	<b>Literaturverzeichnis</b>	<b>91</b>

# Kapitel 1

## Einleitung

### 1.1 Motivation

Die automatisierte maschinelle Erkennung von Personen ist ein gut erforschter Bereich der Informatik und findet in vielen Bereichen Anwendungen. Diese reichen von Mensch-Maschine-Interaktion über Fahrerassistenzsysteme und autonomen Fahrzeugen bis hin zur Überwachung sicherheitskritischer Bereiche. Speziell in der Service- und Assistenzrobotik, wie sie auch am FG NIKR<sup>1</sup> erforscht wird, ist eine sichere Erkennung von Personen notwendig, um Handlungen des Roboters zu planen und auf den Menschen anzupassen. Unter dem Gesichtspunkt, dass keine perfekten Methoden existieren die Personen sicher in allen Situationen erkennen, werden in der Praxis meist unterschiedliche Verfahren miteinander kombiniert. Wobei verschiedene Sensoren zum Einsatz kommen. Ein häufig genutzter Ansatz hierbei ist, die Detektion der spezifischen Signaturen eines Menschen im Scan eines Laser-Entfernungsmessers. Aufgrund diverser Anwendungsmöglichkeiten, besitzen viele autonome Roboter solche Laserscanner auf relativ geringer Höhe. Deshalb wurden in der Vergangenheit einige Verfahren entwickelt, die speziell Beine in Laserscans erkennen und aus ihnen eine Personenhypothese generieren. Diese Masterarbeit beschäftigt sich mit einem neuen Ansatz, wie Personen in solch niedrig positionierten Lasersensoren erkannt werden können. Das Hauptaugenmerk liegt dabei auf der Einsetzbarkeit in Krankenhäusern und Al-

---

<sup>1</sup>Fachgebiet für Neuroinformatik und Kognitive Robotik der TU Ilmenau

tenheimen. Hier scheitern alle bisher entwickelten Ansätze, da in diesen Umgebungen Personen ein teilweise oder komplett anderes Aussehen im Laserscan besitzen können.

## 1.2 Problemstellung

Auch auf den Robotern des Fachgebietes werden kombinierte Verfahren mit unterschiedlichen Sensoren verwendet, um Personen in der näheren Umgebung zu lokalisieren (siehe Abbildung 1.1). Ebenso kommt ein Detektormodul zum Einsatz, dass speziell Beine von nicht-menschlichen Objekten unterscheidet. Da dieses Modul ein trainiertes Verfahren verwendet, hängt die Güte der Klassifikation stark von den verwendeten Daten ab, die für das Training verwendet werden. Hier besteht noch Verbesserungspotential. Weiterhin besteht bisher die Einschränkung, dass nur einzelne Beine erkannt werden können, die sich im Laserscan klar separieren lassen. In der Realität können Personen aber mit weitaus mehr Variation in Erscheinung treten. So existieren Situationen bei denen sich die beiden Beine sehr nahe kommen und somit im Scan nicht mehr getrennt werden können. Desweiteren können sie sich auch teilweise oder ganz verdecken. Betrachtet man speziell Situationen in Krankenhäusern und Altenheimen, kommen zu diesen Erscheinungsformen noch weitere hinzu. So können hier häufig Beine von Krücken oder Rollatoren verdeckt werden oder eine Person in einem Rollstuhl ein komplett anderes Erscheinungsbild besitzen.

## 1.3 Bedeutung für Projekte am Fachgebiet

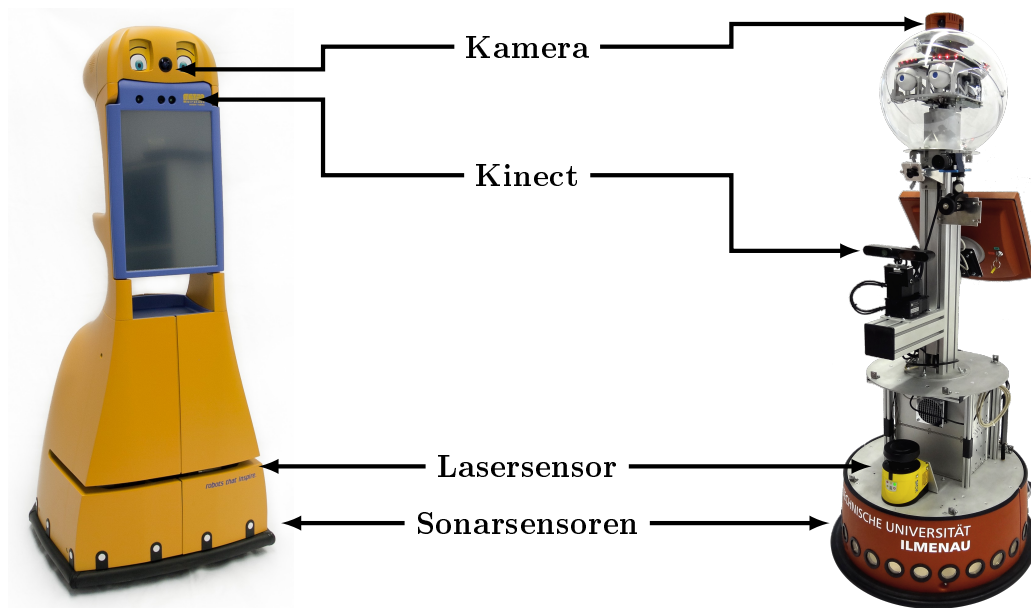
Diese Masterarbeit wurde im Rahmen zweier Projekte erstellt und mit deren spezifischen Einsatzumgebungen evaluiert. Erstens des vom Thüringer Ministerium für Wirtschaft, Arbeit und Technologie geförderten Projekts Projekt SERROGA<sup>2</sup> (**SER**vice-**RO**botik für die **G**esundheits-**A**ssistenz). Sowie des vom Bundesministerium für Bildung und Forschung geförderten Projekts ROREAS<sup>3</sup> (Interaktiver **RO**botischer **RE**ha-**AS**sistent für das Lauf- und Orientierungstraining von Patienten nach Schlag-

---

<sup>2</sup><http://www.tu-ilmenau.de/neurob/projects/serroga/>

<sup>3</sup><http://www.tu-ilmenau.de/neurob/projects/roreas/>

---



**Abbildung 1.1:** Zwei Roboter des Fachgebiets und deren Sensorik

Mit den Lasersensoren der beiden Roboter wurden die Experimente für diese Masterarbeit durchgeführt. **Links** : Tweety aus dem Projekt SERROGA. **Rechts** : Cora aus dem Projekt ROREAS.



anfällen).

Das Projekt SERROGA beschäftigt sich mit der Entwicklung eines “Gesundheitsroboters” und soll in der häuslichen Umgebung älterer Menschen eingesetzt werden, die leichte kognitive Behinderungen besitzen. Der Roboter soll als Erinnerungshilfe für Medikamente und Termine dienen, sowie als Kommunikationsassistent um Kontakt zu Verwandten oder Pflegepersonal aufzunehmen. Desweiteren soll auch die Unversehrtheit überwacht und bei Gefahrensituationen wie stürzen automatisch Hilfe gerufen werden.

Das Projekt ROREAS beschäftigt sich mit der Entwicklung eines “Rehabilitationsroboters” und soll in einem Krankenhaus eingesetzt werden um Schlaganfallpatienten bei der Genesung zu unterstützen. Ein neuer Trend in der Rehabilitation, mit großen medizinischen und ökonomischen Potential, ist das sogenannte Selbsttraining. Hierbei werden Patienten dazu ermutigt neben betreuten Rehabilitationsmaßnahmen selbst aktiv zu werden um ihre vorkrankheitlichen Fähigkeiten wieder zu erlangen. Eine große Rolle spielt hierbei das Lauftraining, welches Krankheitsbedingt oft mit gewissen Ängsten der Patienten korrespondiert. Häufige Gründe warum diese Methode versagt, sind Unsicherheiten bei den Fragen ob sie überhaupt in der Lage sind es durchzuführen oder ob sie selbstständig den Weg zurück in ihr Krankenzimmer finden. Hier soll ein Roboter Hilfestellung leisten, indem er einerseits den Patienten ermöglicht ihr Zimmer wiederzufinden und andererseits den Trainingsfortschritt überwacht und hilft ihn zu verbessern.

Diese beiden Projekte sind stark auf eine besonders natürliche Mensch-Maschine-Interaktion ausgelegt und benötigen somit eine sichere Erkennung von Personen in der Umgebung. Die in dieser Masterarbeit erzeugten Ergebnisse sollen dazu dienen diese zu verbessern, um einerseits die Akzeptanz der Roboter für das menschliche gegenüber zu erhöhen und andererseits die gestellten Projektaufgaben zu erfüllen.

## 1.4 Zielsetzung

Ziel dieser Masterarbeit ist die Verbesserung des am Fachgebiet verwendeten Personendetektors, der mittels Klassifikation von Laserscandaten Personenhypothesen ge-

---

neriert. Hierzu soll zuerst eine geeignete Menge von Trainings- und Testdaten aufgenommen, die Probleme des zurzeit verwendeten Detektionsmoduls analysiert und eine Evaluierung auf Realwelt-Testszenarien durchgeführt werden. Des Weiteren sollen neue Merkmale implementiert und evaluiert werden, die eine allgemeinere Personenhypothese zulassen. Diese sollen sich nicht nur auf Beine beschränken, sondern auch Personen mit Krücken bzw. Rollatoren oder in Rollstühlen sicher beschreiben können.

## 1.5 Überblick

Der Rest dieser Arbeit ist wie folgt gegliedert. In Kapitel 2 wird ein Überblick gegeben, welche Verfahren für Personendetektion dem derzeitigen Stand der Wissenschaft entsprechen. Dazu wird zuerst auf Personendetektion im Allgemeinen und danach speziell in Lasermessungen eingegangen. In Kapitel 3 wird zuerst auf einige Allgemeine Probleme der Personendetektion in Laserscans eingegangen, sowie das Verfahren erläutert, welches zurzeit am Fachgebiet eingesetzt wird. Kapitel 4 beschäftigt sich mit den Merkmalen, die für diese Masterarbeit neu implementiert wurden. Hierzu wird auf die Grundidee sowie auf den Ablauf und die Unterschiede zum bestehenden Ansatz eingegangen. In Kapitel 5 werden die Ergebnisse präsentiert, die zuerst einen Vergleich erlauben sollen wie gut die Verfahren jeweils für die Personendetektion durch das Erkennen von Beinen geeignet ist. Danach wird dargelegt, inwieweit es möglich ist auch Personen mit abweichenden Erscheinungsformen zu erkennen. In Kapitel 6 wird schlussendlich ein Resümee gezogen und erläutert, wie das neue Verfahren in der Zukunft möglicherweise verbessert werden könnte.

---



## Kapitel 2

### State of the Art

Das maschinelle automatisierten Erkennen von Personen in der Umgebung, ist ein gut erforschtes Problem in der Informatik, für das eine Vielzahl von Verfahren entwickelt wurde. Wichtige Einsatzszenarien sind zum Beispiel die Überwachung sicherheitskritischer Bereiche wie Flughäfen oder Fußballstadien. Auch für die Sicherstellung menschlicher Unversehrtheit bei zum Beispiel industriellen Robotern oder autonomen Fahrzeugen ist dieses Problem von grundlegender Bedeutung. In der Service- und Assistenzrobotik ist die sichere Erkennung von Personen eine der wichtigsten Grundlagen um eine besonders natürliche Mensch-Maschine-Kommunikation zu ermöglichen. Eng verbunden mit dem Problem der Detektion von Personen, ist das sogenannte Tracking. Hierbei handelt es sich um die Fähigkeit, durch Detektion erstellte Personenhypothesen über längere Zeitabschnitte zu verfolgen. Dies ist insofern von Bedeutung, da alle bekannten Detektionsverfahren spezifischen Einschränkungen unterliegen, hinsichtlich Umwelteinflüssen und Berechnungskomplexität. Hierzu werden in der Praxis meist mehrere Detektionsverfahren auf unterschiedlichen Sensoren parallel eingesetzt, die ihre Hypothesen an ein Trackingmodul übergeben. Somit ist das Tracking von großer Bedeutung für einen Roboter, damit er nicht “vergisst” wo sich eine Person befindet, auch wenn zeitweise keine Detektionen vorliegen. Diese Masterarbeit beschäftigt sich mit dem reinen Detektionsproblem von Personen in Laserscans, es sollte aber immer im Auge behalten werden, dass dies in der Praxis immer Teil eines größeren Gesamtsystems ist.

---

## 2.1 Sensorik für Personendetektion auf mobilen Robotersystemen

### 2D Kameras

Einer der beliebtesten Sensoren für die Personendetektion auf mobilen Robotersystemen, sind konventionelle 2D Kameras. Aufgrund relativ geringer Kosten, ist diese Sensorenart heutzutage auf den meisten Robotersystemen vorhanden. Für diese Sensoren wurden eine Reihe von Verfahren entwickelt, die spezifische Merkmale von Personen erkennen, wie Konturen [ZHOU und HOANG, 2005] oder Gesichter [VIOLA und JONES, 2001]. Aufgrund des hohen Informationsgehaltes dieser Sensoren, können Personenhypothesen relativ sicher generiert werden, was allerdings auch zu einem erhöhten Berechnungsaufwand führt. Ein Nachteil dieser Sensoren ist, dass diese je nach Position am Roboter nur über einen relativ eingeschränkten Sichtbereich verfügen. So können Personen, die zu weit seitlich oder zu nah an der Kamera stehen, ganz oder teilweise aus ihren Messbereich fallen. Desweiteren sind die meisten Verfahren sehr empfindlich gegenüber Beleuchtungsänderungen. Ein weiterer Nachteil bezüglich der Generierung der Personenhypothese ist, dass 2D Kameras keine Tiefeninformationen liefern und somit die Entfernung der Person zur Kamera nur grob abschätzbar ist (siehe Abbildung 2.1).

### Lasersensoren

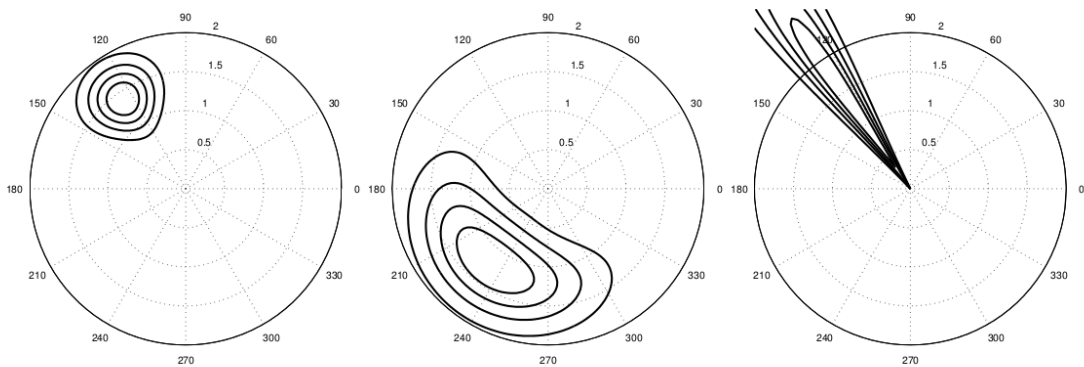
Lasersensoren, im Englischen meist abgekürzt durch LIDAR (**L**ight **d**etection **a**nd **r**anging) oder LADAR (**L**aser **d**etection **a**nd **r**anging), sind eine weitere Sensorenart für die Personendetektion. Aufgrund vielseitiger Einsatzmöglichkeiten, sind diese auf den meisten Robotersystemen vorhanden. Lasersensoren liefern im Gegensatz zu Kameras sehr genaue Entfernungsmessung und können deshalb den Standpunkt von Personen sehr Präzise bestimmen. Desweiteren besitzen sie einen sehr weiten Messbereich und sind unempfindlich gegenüber Beleuchtungsänderungen. Deshalb werden Kameras und Lasersensor oft zusammen verwendet. Nachteile dieser Sensoren sind der relativ geringe Informationsgehalt und der daraus folgenden relativ hohen

---

Rate von falschen Detektionen.

### weitere Sensoren

Desweiteren wurde eine Fülle von weiteren Verfahren entwickelt, die andere Sensoren verwenden. So wurden 3D Kameras verwendet, um Tiefensignaturen von Personen im Raum zu erkennen [IKEMURA und FUJIYOSHI, 2011]. Wobei dieses Verfahren stark von einer guten Vorverarbeitung und der Personenpose abhängig ist (laut [CAMPANA, 2011]). Infrarotkameras wurden genutzt um Wärmesignaturen von Menschen zu erkennen [TREPTOW et al., 2006]. Allerdings besitzt diese Methode ebenfalls Nachteile wenn mehreren Personen sich gegenseitig verdecken. Stereomikrofone wurden verwendet um menschliche Geräuschquellen zu identifizieren [FRITSCH et al., 2004]. Wobei in dieser Arbeit angenommen wurde, dass nur die Person spricht, die mit dem Roboter interagiert. Auch Sonarsensoren wurden für die Detektion von Personen verwendet [MARTIN et al., 2006], obwohl diese nur sehr ungenaue Personenhypothesen zulassen.



**Abbildung 2.1:** Beispiele für Personenhypothesen von drei unterschiedlichen Sensoren.

*Der Roboter befindet sich jeweils im Zentrum der Grafiken. **Links** : Lasersensor mit Personenhypothese nordwestlich. **Mitte** : Sonarsensor mit Personenhypothese südwestlich. **Rechts** : 2D Kamera mit Personenhypothese nordwestlich. (Abbildung übernommen aus [MARTIN et al., 2006])*

Zusammenfassend zu diesem Abschnitt ist also festzuhalten, dass eine starke und gegenüber Umwelteinflüssen robuste Personenhypothese zur Zeit nur durch die Kombination verschiedener Sensoren erreicht werden kann. Der nächste Abschnitt beschäftigt sich mit den konkreten Verfahren, welche speziell für Lasersensoren entwickelt wurden.

## 2.2 Personendetektion in Laserscans

Forschungsarbeiten die sich mit der Detektion von Personen in Laserscans beschäftigen, können grundsätzlich in zwei Fachgebiete eingeteilt werden. Dem Gebiet der intelligenten Fahrzeuge und dem Gebiet der mobilen Robotik. Diese beiden Fachbereiche weisen teilweise starke Unterschiede bezüglich ihrer Einsatzumgebungen auf. In der mobilen Robotik sind die Lasermessungen aufgrund der indoor Einsatzumgebung meist auf kürzere Distanzen beschränkt als es bei den intelligenten Fahrzeugen der Fall ist. Deshalb können hier Lasersensoren mit geringerer Auflösung eingesetzt werden. Desweiteren kann in der mobilen Robotik die Annahme getroffen werden, dass sich nur Personen in aufeinanderfolgenden Messungen bewegen und der Rest der Objekte statisch an einem Platz verweilt. Bei den intelligenten Fahrzeugen hingegen können auch nichtmenschliche Objekte, wie andere Fahrzeuge, sich bewegen. Trotz dieser Unterschiede können die entwickelten Verfahren aber meistens auch in dem jeweils anderen Fachgebiet eingesetzt werden.

### 2.2.1 2D Verfahren

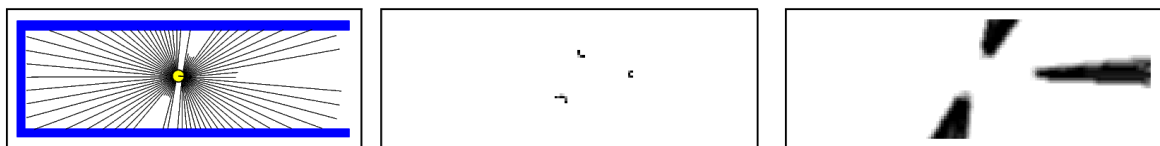
Die ersten Verfahren, die sich mit dem Erkennen von Personen in Laserscans beschäftigten, waren sehr auf das nachfolgende Tracking fokussiert. So wurden für die reine Erkennung in Lasermessungen zuerst meist nur einfache Ansätze vorgeschlagen, die häufig großen Einschränkungen unterlagen.

In [FOD et al., 2002] wurde ein Verfahren beschrieben, das mittels fest stehender

---

Lasersensoren Personen in der Umgebung Trackt. Dazu extrahierten sie Segmente aus den Laserscans (hier noch Blobs genannt). Diese Segmente sind nah beieinanderliegenden Messpunkten, welche eine kontinuierliche Oberfläche bilden und somit zu einem Objekt im Raum gehören. Für die Erkennung von Personen wurde zunächst ein Modell der Umgebung gelernt. In der Anwendungsphase kann anhand dieser Informationen festgestellt werden, welche Segmente von Personen erzeugt werden und welche zur Umgebung gehören. Die eigentliche Aufgabe des Verfahrens bestand nun darin, festzustellen wie sich die Segmente von Personen in aufeinanderfolgenden Laserscans bewegt haben. Dadurch konnte mittels eines Bewegungsmodells und eines extendet Kalman Filters festgestellt werden welche dieser Segmente zu den einzelnen Person gehören. Obwohl dieses Verfahren nicht auf mobilen Robotersystemen anwendbar ist, wurde das Konzept der Segmentierung für nahezu alle weiteren Verfahren beibehalten.

Für mobile Roboter wurde in [SCHULZ et al., 2003] ein ähnlicher Ansatz Vorgestellt. Wobei angenommen wurde, dass Personen im Laserscan immer lokale Minima darstellen (siehe Abbildung 2.2 und 2.3). Dieses Verfahren benötigte kein vortrainiertes Umgebungsmodell, sondern erstellte anhand aufeinanderfolgender Laserscans eine lokale Karte. Mittels Scanmatching können nun neue Lasermessungen zu der Karte ausgerichtet werden. Durch Differenzbildung ist es dann möglich sich bewegende Segmente zu erkennen, welche wiederum als Personen angenommen wurden. Im Gegensatz zu Fod et. al verwendeten sie einen Particle Filter für das Tracking.

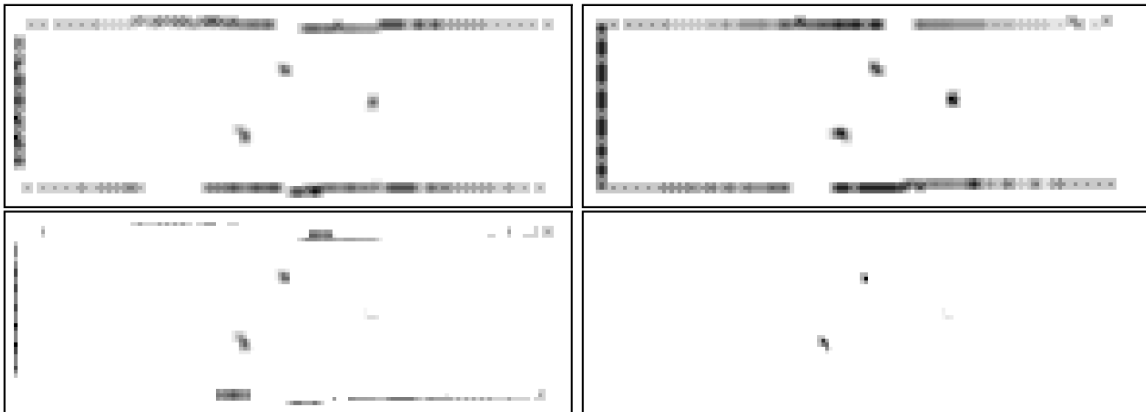


**Abbildung 2.2:** Laserscan mit lokalen Minima.

**Links :** der Laserscan. **Mitte :** Die lokalen Minima der Feature Map. Diese wurden durch zwei Personen und einem Mülleimer (rechtes Minima) erzeugt. **Rechts :** Okklusionskarte. (Abbildung übernommen aus [SCHULZ et al., 2003]).

Die Grundidee, Personen als sich bewegende lokale Minima im Laserscan zu betrach-





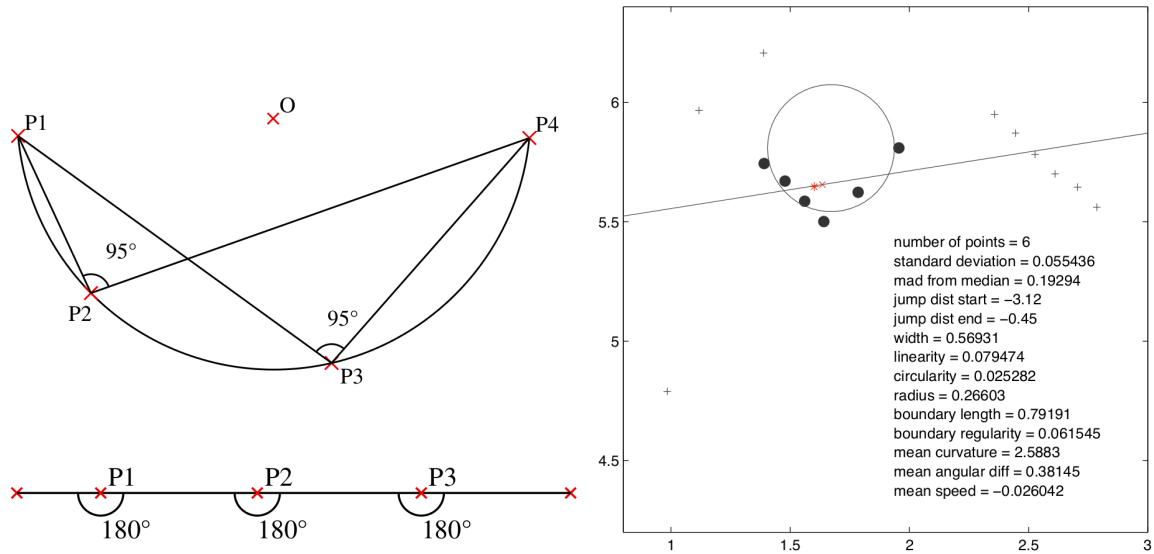
**Abbildung 2.3:** Occupancy maps für die Erkennung bewegter Objekte.

***Oben links :** die Karte vom vorherigen Zeitschritt. **Oben rechts :** die Karte vom derzeitigen Zeitschritt. **Unten links :** die Differenz beider Karten. **Unten rechts :** die Fusion beider Differenzkarte mit der Feature Map aus Abbildung 2.2 (Abbildung übernommen aus [SCHULZ et al., 2003])*

ten und mittels probabilistischer Filter zu Tracken, wurde dabei in den nächsten Jahren zu einem der beliebtesten Ansätze um das Trackingproblem zu behandeln. Abwandlungen hiervon können für statische Lasersensoren in [CUI et al., 2005], für mobile Roboter in [BENNEWITZ et al., 2005], gefunden werden. Ein großes Problem aller dieser Methoden ist allerdings, dass mit ihnen nur sich bewegend Personen erkannt werden können und solche die an einer Stelle verweilen ignoriert werden.

In [KLEINEHAGENBROCK et al., 2002] wurde ein Verfahren vorgestellt, welches die spezifischen geometrischen Merkmale von Beinen in Laserscans verwendete. Somit war es das Erste, welches in der Lage war ohne Hintergrundmodell still stehende Personen zu erkennen. Dafür wurde der Laserscan ebenfalls in Segmente eingeteilt und dann anhand nicht näher erläuteter Thresholds entschieden ob ein Segment als Bein oder nicht gewertet werden soll. Auch hier lag das Hauptaugenmerk auf der Fusion von Personenhypothesen aus Kamerabildern und Laserscans sowie dem Tracking. Aufgrund relativ einfacher Testszenarien und der Einschränkung, dass nur Segmente in einem Winkel von  $60^\circ$  und einem Abstände von  $150\text{cm} \pm 50\text{cm}$  zur Front

des Roboters überhaupt als Hypothese in Frage kommt, muss dieses Verfahren als konzeptioneller Ansatz gewertet werden.



**Abbildung 2.4:** Merkmale für Beinsegmente.

**Links :** eines der von Xavier et. al. verwendeten Merkmale. Die inneren Winkel eines kreisähnlichen Segments sind deutlich geringer als die einer Linie. (Abbildung übernommen aus [XAVIER et al., 2005]). **Rechts :** die 14 von Arras et. al. verwendeten Merkmale. (Abbildung übernommen aus [ARRAS et al., 2007]).

In [XAVIER et al., 2005] wurde eine weitere Methode beschrieben, die versuchte die Eigenschaften von Beinen geometrisch zu beschreiben. Dazu definierten sie Merkmale, welche mathematische Modelle beschreiben (Linien und Kreise siehe Abbildung 2.4 links). Anhand manuell festgelegter thresholds wurde dann entschieden, ob Segmente ein Bein sind oder zum Hintergrund gehören. Dieses Verfahren war eigentlich dazu gedacht Beine aus Laserscans zu entfernen. Dadurch sollte das von ihnen verwendete SLAM<sup>1</sup> Verfahren verbessert werden. Dennoch kann diese Methode als erster praktikabler Ansatz gewertet werden, Personen in Laserscans durch ihre spezifischen geometrischen Merkmale zu erkennen und nicht nur sich bewegende Segmente als diese anzunehmen. Durch die manuelle Festlegung der thresholds, muss die Effizienz

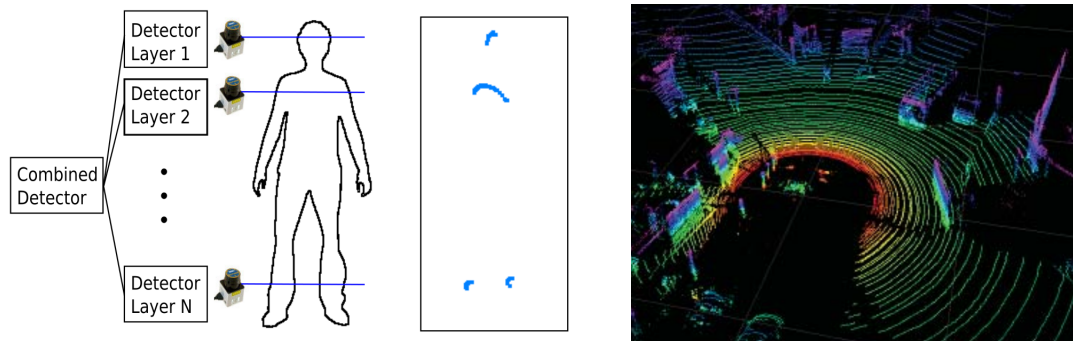
<sup>1</sup>Simultaneous Localisation and Mapping

dieses Verfahrens aber dennoch in Frage gestellt werden.

In [ARRAS et al., 2007] wurde das Detektionsproblem erstmals unabhängig vom Tracking der Person betrachtet. Die Motivation war dabei, dass sich frühere Publikation mehr mit der Entwicklung und Verbesserung von Tracking Algorithmen beschäftigten und dabei die Definition und Analyse von geeigneten Merkmalen vernachlässigt wurde. Dafür wurden 14 geometrische und statistische Merkmale, welche die Eigenschaften von Beinen gut beschreiben sollten definiert. Zusätzlich wurde noch ein Merkmal das auf Bewegung basiert hinzugenommen. Für die Analyse welche dieser Merkmale am besten dafür geeignet sind ein Bein zu beschreiben, verwendeten sie das maschinelle Lernverfahren Adaboost [FREUND und SCHAPIRE, 1995], welches im Bereich der Personendetektion durch das Erkennen von Gesichtern schon herausragende Ergebnisse erzielte [VIOLA und JONES, 2001]. Adaboost ist dabei in der Lage einen Klassifikator zu trainieren und gleichzeitig die Merkmale zu ermitteln, welche besonders deskriptiv sind. Diese Methode erfuhr große Beliebtheit und die aus dieser Arbeit hervorgegangenen Merkmale wurden noch in weiteren Ansätzen verwendet und abgewandelt [PREMEBIDA et al., 2009, KONDAXAKIS et al., 2009, ZIVKOVIC und KROSE, 2007].

---

### 2.2.2 3D Verfahren



**Abbildung 2.5:** Sensorik für Personendetecktion in 3 Dimensionen.

*Links :* Anordnung von Lasersensoren auf unterschiedlichen höhen und korrespondierende Scanpunkte (Abbildung übernommen aus [MOZOS et al., 2009]).

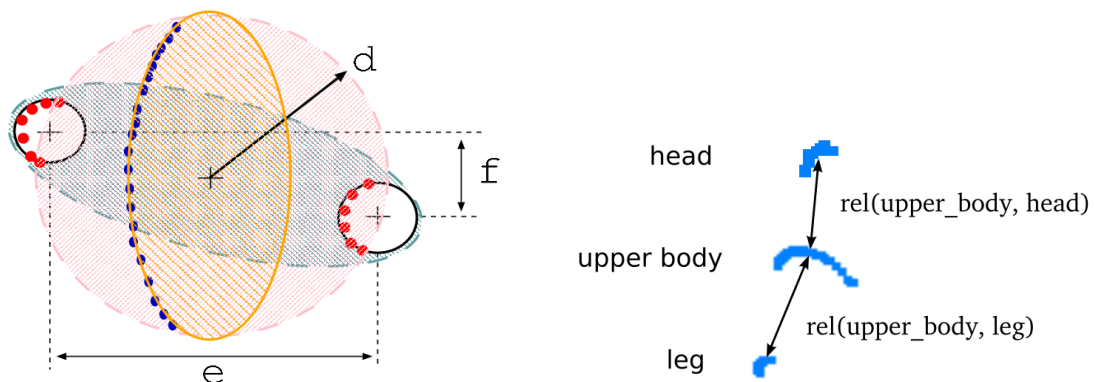
*Rechts :* Laserscan eines 3D Laserscanners (Abbildung übernommen aus [KIDONO et al., 2011])

Neuere Verfahren, die sich mit der Detektion von Personen in Laserscans beschäftigen, behandeln hauptsächlich Methoden wie diese im 3 Dimensionalen Raum erkannt werden können. Um die Daten zu generieren, können hierzu entweder konventionelle 2D Lasersensoren auf unterschiedlichen Höhen angebracht werden oder es werden 3D Lasersensoren verwendet, die den Raum in unterschiedlichen vertikalen Scanwinkeln messen (siehe Abbildung 2.5).

In [ALEXANDER CARBALLO und YUTA, 2008] wird eine Methode vorgestellt, wie Personen mit Hilfe von zwei Lasersensoren auf unterschiedlichen Höhen detektiert werden können. Hierzu wurde ein 2D Sensor auf Beinhöhe und ein anderer auf Oberkörperhöhe positioniert. Merkmalsgenerierung und Klassifikation entsprechen hierbei größtenteils dem von Arras vorgestellten Ansatz, nur das für den oberen Sensor elliptische Merkmale verwendet werden. Die einzelnen Schichten werden dann separat klassifiziert und zu einer Personenhypothese fusioniert. Diese Fusion erfolgte dabei indem sie um die Ellipse, die durch den Oberkörper gebildet wurde, einen Kreis legten und in dessen Radius die korrespondierenden Beine der Person suchen (siehe

Abbildung 2.6 links).

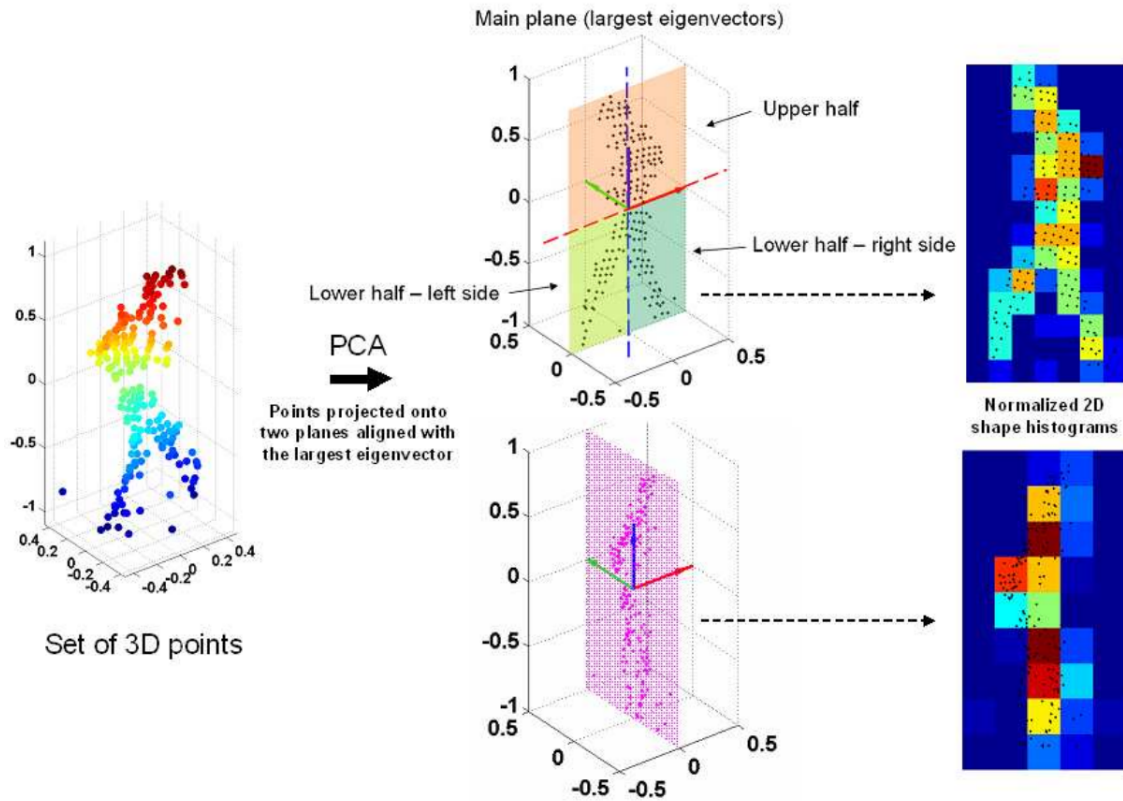
In [MOZOS et al., 2009] wurde ein Verfahren beschrieben, wie Personen mit beliebig vielen 2D Lasersensoren auf unterschiedlichen Höhen detektiert werden können (siehe Abbildung 2.5 links). Für die Klassifikation der Segmente auf den einzelnen Schichten wurde ein subset von 11 Merkmalen verwendet, die Arras et. al. vorgestellt hatten. Für die Fusion der einzelnen Schichten lernten sie ein Shape Model, das die geometrischen Relationen (Euklidischer Abstand) zwischen den einzelnen positiv klassifizierten Segmenten lernt und anhand einer gewichtete Mehrheitsentscheidung festlegt, ob es sich um eine Person handelt oder nicht (siehe Abbildung 2.6 rechts).



**Abbildung 2.6:** Modelle zur Fusion von mehreren Sensorebenen.

**Links :** Modell zur Fusion zweier Schichten von Carballo. Im Umkreis der Oberkörperellipse (orange), werden auf der unteren Scanschicht Beine gesucht. Falls diese gefunden wurden, wird der Mittelpunkt der Ellipse als Personenhypothese verwendet (Abbildung übernommen aus [ALEXANDER CARBALLO und YUTA, 2008]). **Rechts :** Shape Model von Mozos et. al. anhand der geometrischen Relationen der positiv klassifizierten Segmente unterschiedlicher Schichten, wird entschieden ob es sich um eine Person handelt (Abbildung übernommen aus [MOZOS et al., 2009]).

In [NAVARRO-SERMENT et al., 2010] wurde ein Verfahren vorgestellt, welches einen 3D Laserscanner verwendet. Hierzu werden Punktwolken aus dem Laserscan isoliert, welche eine bestimmte Höhe über der Bodenebene besitzen. Diese kommen als möglich

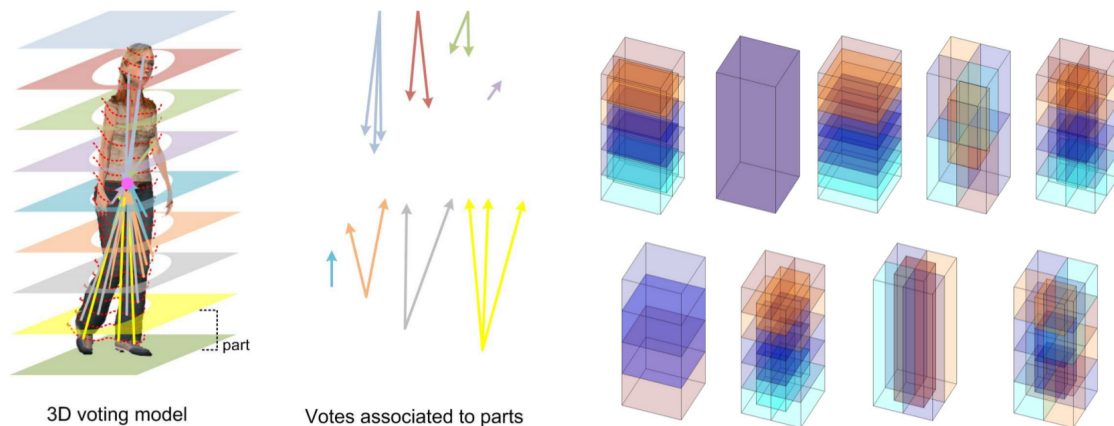


**Abbildung 2.7:** Histogramme aus 3D Laserdaten.

*Links : Punktwolke des Laserscans. Mittels einer Hauptkomponentenanalyse, werden die Punkte in zwei Unterräume projiziert. Mitte oben : Unterraum des ersten und zweiten Eigenvektors. Mitte unten : **Rechts** : Aus den Ebenen werden Histogramme gebildet, welche als Merkmalsvektor dienen. Von blau nach rot ansteigende Punktedichte. (Abbildung übernommen aus [NAVARRO-SERMENT et al., 2010])*

Kandidaten für Personen in Frage. Für die Merkmalsgenerierung werden Histogramme aus den Messdaten erzeugt. Die Punktwolken werden dabei mittels einer Hauptkomponentenanalyse in die Ebenen des ersten und zweiten bzw. ersten und dritten Eigenvektors projiziert. Diese beiden Ebenen werden nun in Bins eingeteilt und anhand der Punktdichte innerhalb der Bins normalisierte 2D Histogramme gebildet (siehe Abbildung 2.7). Für die Klassifikation verwenden sie Support Vector Maschinen.

In [SPINELLO et al., 2010] wurde ein Verfahren vorgestellt, welches einzelne horizon-



**Abbildung 2.8:** Zweistufiges Detektionssystem von Spinello.

**Links :** die Parts umfassen alle Punkte, die zwischen den eingezeichneten Ebenen gemessen werden. Die Votes jeder dieser Parts werden verwendet um Personenhypothesen im 3D Hypothesenraum zu erzeugen. **Rechts :** Bounding Boxen die um bestehende Personenhypothesen generiert werden, um diese zu validieren. (Abbildungen übernommen aus [SPINELLO et al., 2010] und [SPINELLO et al., 2011]).

tale Scanlinien aus einem 3D Scan verwendet um Personen zu detektieren. Dieses Verfahren nutzt die Eigenschaft von 3D Laserscannern, dass der Raum ebenweise mit unterschiedlichen horizontalen Scanwinkeln gemessen wird. Horizontal benachbarte Scanebenen werden dabei zu sogenannten Parts zusammengefasst (siehe Abbildung 2.8 links). Für jeden Part wird daraufhin ein Adaboost Klassifikator trainiert. Innerhalb der Parts werden dazu die Messpunkte einer Scanebene in Segmente unterteilt, aus denen 17 geometrische und statistische Merkmale berechnet werden. Die separate Klassifizierung unterschiedlicher höhen soll dabei die Robustheit gegenüber Verdeckungen verbessern. Für die Generierung der Personenhypothese wird ein Votingmodell verwendet. Hierzu werden in der Trainingsphase Differenzvektoren der Segmente zum Schwerpunkt der Person berechnet und mittels Clustering zu 'Votes' für einen Klassifikator zusammengefasst. In der Anwendungsphase werden diese Votes wieder in den Hypothesenraum projiziert. Regionen mit einer hohen Dichte an Votes werden dann als Schwerpunkt der Personenhypothese verwendet. Dieses Verfahren wurde in [SPINELLO et al., 2011] um einen nachfolgenden Schritt erweitert.

Die Motivation hierfür bestand darin, dass der bestehende Ansatz bei Szenen mit mehreren Personen oft Personenhypothesen generierte, welche zwischen den Personen liegen. Deshalb werden 3D Bounding Boxen um die im ersten Schritt generierten Personenhypothesen erzeugt. Innerhalb dieser werden viele unterschiedliche sich teilweise überlappende Subboxen erzeugt (siehe Abbildung 2.8 rechts). Durch die Messpunkte innerhalb dieser Subboxen werden nun 9 Merkmale generiert. Auch hier wird ein Adaboost Klassifikator verwendet, der zuvor erzeugte Personenhypothesen bestätigt oder verwirft.

### Zusammenfassung

Autoren	Kritikpunkte
[FOD et al., 2002]	nicht anwendbar auf mobilen Robotern
[KLEINEHAGENBROCK et al., 2002]	konzeptioneller Ansatz
[SCHULZ et al., 2003]	erkennt nur sich bewegende Objekte als Personen
[XAVIER et al., 2005]	manuelle Parametersetzung für die Klassifikation
[ARRAS et al., 2007]	State of the Art für einzelne 2D Lasersensoren
[ALEXANDER CARBALLO und YUTA, 2008]	benötigt zwei 2D Lasersensoren
[MOZOS et al., 2009]	benötigt mehrere 2D Lasersensoren
[NAVARRO-SERMENT et al., 2010]	benötigt 3D Lasersensor
[SPINELLO et al., 2011]	benötigt 3D Lasersensor

**Tabelle 2.1:** Übersicht und Kritikpunkte zu den Verfahren der State of the Art Recherche.

*Der von Arras et. al. verwendete Ansatz (grün) wurde als das fortschrittlichste Verfahren für einzelne 2D Lasersensoren identifiziert und wird auch am Fachgebiet verwendet.*



In diesem Abschnitt wurden verschiedene Verfahren vorgestellt, wie Personen in Laserscans erkannt werden können. Diese verwendeten entweder einen oder mehrere 2D Lasersensoren oder einen 3D Lasersensor. Aus Kostengründen verfügen viele sich heute am Markt befindliche Roboter allerdings nur über einen einzelnen 2D Lasersensor. So auch die am Fachgebiet eingesetzten Roboter. Das fortschrittlichste Verfahren, welches mit 2D Laserdaten arbeitet, war dabei das von Arras et. al. beschriebene. Dieses verwendet geometrische und statistische Merkmale, um Beine von Personen zu erkennen. Im nächsten Kapitel wird dieses Verfahren genauer vorgestellt, sowie die konkret am Fachgebiet vorhandene Implementierung erläutert.

---

# Kapitel 3

## Grundlagen

Da sich alle State of the Art Verfahren für die Personendetektion in 2D Laserscans auf die Erkennung von Beinen beschränken, soll in diesem Kapitel zunächst auf die grundlegende Arbeitsweise dieser Methoden eingegangen werden. Hierzu werden zunächst einige verfahrensunabhängige Probleme diskutiert, welche auf die Qualität eines Personendetektors Einfluss nehmen. Im zweiten Abschnitt wird der Ablauf einer Personendetektion vorgestellt, wobei die essentiellen Schritte der Merkmalsgenerierung und Klassifikation dem Verfahren von Arras et. al. entsprechen. Dieses wurde im vorherigen Kapitel als das Fortschrittlichste identifiziert und dient somit als Referenz für die in Kapitel 4 vorgestellte Methode. Am Ende dieses Kapitels werden noch einige Methoden erläutert, wie im allgemeinen die Personendetektion durch die Optimierung des verwendeten Klassifikators verbessert werden kann.

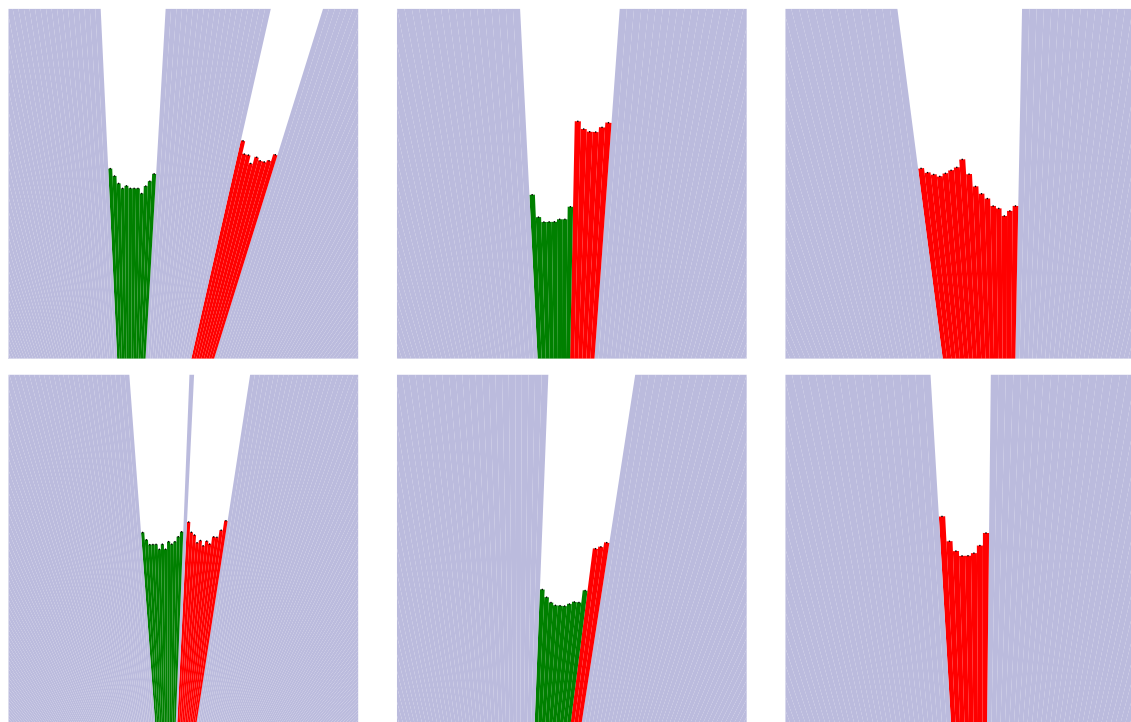
### 3.1 Grundlegende Probleme der Personendetektion

Wie im vorherigen Kapitel erwähnt, werden für die Personendetektion in Laserscans benachbarte Messpunkte im Laserscan zu Segmenten zusammengefasst. Segmente die durch Personen erzeugt werden können dabei je nach Stellung der Beine teilweise sehr unterschiedliche Formen annehmen. So können sich Beine teilweise verdecken oder zu einem Segment zusammengefasst werden, falls sich diese zu nah beieinander befinden. Dies wirft zum einen die Frage auf, wie überhaupt eine Personenhypothese

---

erstellt werden soll. So kann verlangt werden, dass zwei Beine erkannt werden müssen um aus ihnen eine Hypothese zu generieren. Dies erzeugt sehr starke Hypothesen und sorgt für relativ wenige falsche Detektionen. Allerdings werden auch einige potentiell erkennbare Erscheinungsformen kategorisch von einer Detektion ausgeschlossen. Falls auch einzelne Segmente für eine Hypothese verwendet werden, können zwar potentiell alle Erscheinungsformen erkannt werden, jedoch kann dies auch die Anzahl von falschen Detektionen erhöhen. Eine andere Fragestellung behandelt das Problem, dass die unterschiedlichen Erscheinungsformen auch verschieden schwer zu klassifizieren sind. Man steht also ebenso vor der Wahl, mit welchen Segmenten man überhaupt einen Klassifikator trainieren möchte.

### Klassen von Erscheinungsformen



**Abbildung 3.1:** Erscheinungsformen von Beinen in Laserscans

***Links :** klar separierbare Segmente. **Mitte :** teilweise verdeckt. **Rechts :** nicht separierbar oder komplett verdeckt. Verschiedene Segmente durch grün und rot gekennzeichnet.*

Für die Analyse welche Entscheidungen getroffen werden sollten um einen möglichst guten Personendetektor zu erzeugen, wurden für diese Masterarbeit die Segmente in drei Klassen eingeteilt (siehe Abbildung 3.1). Da für einen guten Klassifikator sehr große Datenmengen benötigt werden, geschieht dies Algorithmisch nach folgendem Schema.

Die erste Klasse besteht aus Beinen, die sich im Laserscan klar separieren lassen. Sie wird dadurch definiert, dass die Beinsegmente die zur selben Person gehören mindestens einen Messpunkt zwischen ihnen besitzen der nicht von der Person stammt. Diese Klasse enthält somit nur halbkreisähnliche Beinsegmente und ist am einfachsten zu klassifizieren. Die zweite Klasse besteht aus Beinsegmenten die sich teilweise verdecken. Sie wird dadurch definiert, dass die beiden Segmente zur selben Person gehören und kein Messpunkt zwischen ihnen liegt der nicht zu der Person gehört. Somit fallen in diese Klasse sowohl einfach zu klassifizierende halbkreisähnliche Beinsegmente, als auch verdeckte Segmente, die nicht dem erwarteten Aussehen eines Beines entsprechen. Diese Klasse ist somit mittelschwer zu klassifizieren. Die dritte Klasse von Segmenten besteht aus Beinen die nicht separiert werden können. In diese Klasse fallen Segmente, die entweder aus zwei Beinen bestehen (im Bild rechts unten) oder aus einem einzelnen Bein, welches das korrespondierende vollständig verdeckt (im Bild rechts oben). Durch die hohe Varianz des Aussehens dieser Segmente, ist diese Klasse am schwersten zu klassifizieren.

## 3.2 Ablauf der Personendetektion

Die Personendetektion läuft in dem von Arras beschriebenen Verfahren, sowie im neuen Ansatz grundsätzlich in fünf aufeinanderfolgenden Schritten ab (siehe Abbildung 3.2).



Abbildung 3.2: Die Verarbeitungsschritte einer Personendetektion.

### 3.2.1 Vorverarbeitung

Während der Vorverarbeitungsphase werden die Messdaten von dem Sensor aufgenommen und an das Detektionsmodul übergeben (3.3). Die Messpunkte befinden sich dabei in Polarkoordinaten und werden durch ihre Radialkoordinate  $r$  und ihre Winkelkoordinaten  $\phi$  beschrieben. Diese müssen zunächst in kartesische Koordinaten umgewandelt werden. Diese können mit  $x = r \cdot \cos(\phi)$  und  $y = r \cdot \sin(\phi)$  berechnet werden.

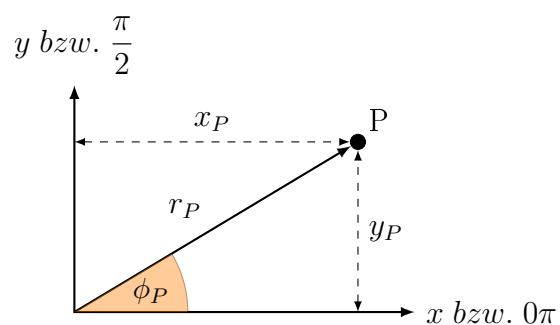
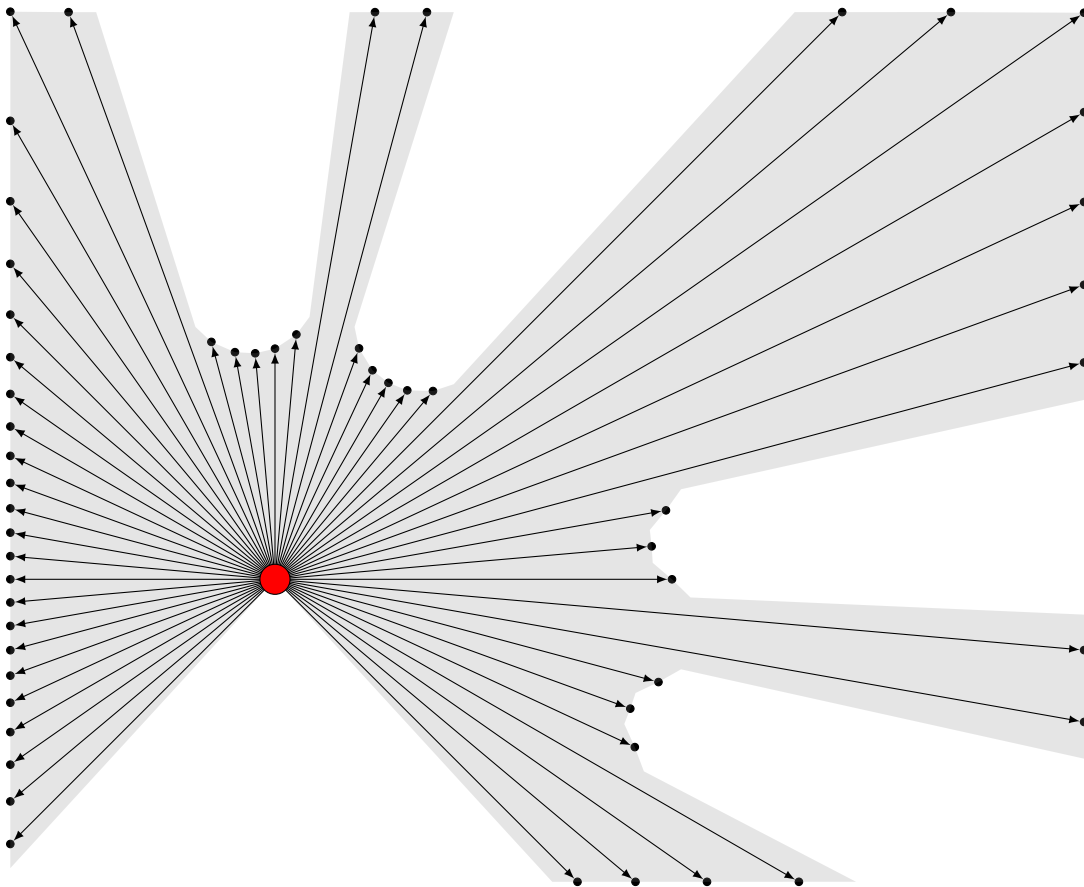


Abbildung 3.4: Geometrische Relationen von kartesischen- und Polarkoordinaten.



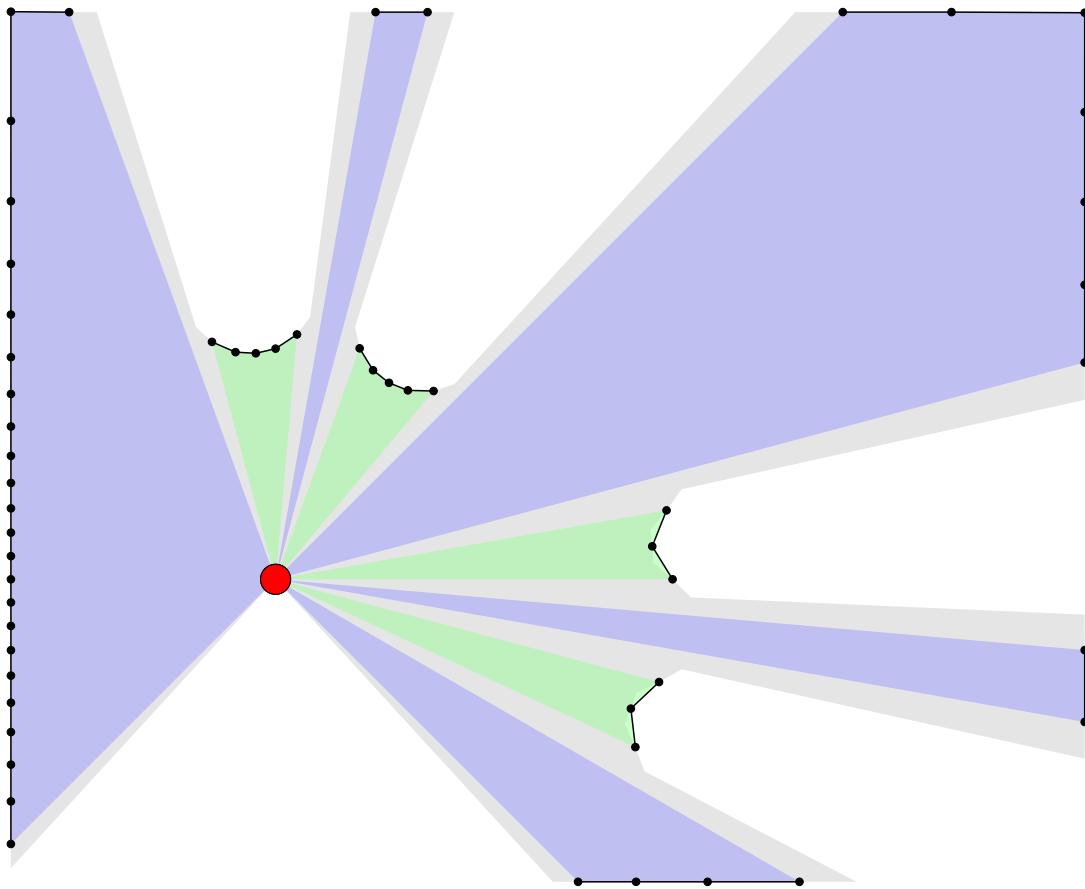
**Abbildung 3.3:** Schematische Darstellung eines 270° Laserscans

*Messpunkte (schwarz), werden durch ihren Abstand zum Roboter (rot) und Winkel ihrer Scanlinie (Pfeile) beschrieben. Die Beine zweier Personen mit unterschiedlichem Abstand befinden sich vor und rechts des Roboters. Messbereich grau.*

### 3.2.2 Segmentierung

Der nächste Schritt ist die Segmentierung. Hierbei wird das Problem behandelt, wie Punkte im Raum zu Gruppen zusammengefügt werden können, sodass möglichst jede dieser Gruppen ein Objekt im Laserscan beschreibt (siehe Abbildung 3.5).

Um dieses Problem zu behandeln stehen mehrere Möglichkeiten zur Verfügung. Ein Überblick kann in [NUNES und PREMEBIDA, 2005] gefunden werden. Ein beliebter Ansatz, der auch von Arras verwendet wurde, ist hierbei die Segmentierung über die



**Abbildung 3.5:** Segmentierter Laserscan.

*Punkte eines Segments sind durch schwarze Linien verbunden, adjazente Segmente im Scanbereich durch unterschiedliche Farben gekennzeichnet.*

Jump Distance (siehe Pseudocode in Abbildung 3.6). Der Algorithmus übernimmt eine Sequenz von nach Scanwinkel geordneten Messpunkten. Falls zwei aufeinanderfolgende Punkte einen größeren Abstand als die festgelegte Jump Distance besitzen, gehören diese nicht mehr zum selben Segment bzw. sie definieren die Endpunkte der Segmente. Als Ausgabe wird wiederum eine nach Winkel geordnete Sequenz von Segmenten erzeugt, welche die Messpunkte beinhalten.

**Eingaben**


---

```

1    $\mathbf{X} = \{x_1, x_2, \dots, x_M\}$  // Punkte des Laserscans
2    $d_{jump}$  // Jump Distance
3    $dist(x_i, x_j)$  // Distanzfunktion

```

**Algorithmus**

```

4   j:=1;
5   initialisiere  $S_j$ ; // initialisiere erstes Segment
6   Für jeden  $p_i : i \in \{1, \dots, M-1\}$ ;
7        $d := dist(x_i, x_{i+1})$ ; // Distanz zweier Messpunkte
8       falls  $d < d_{jump}$ ; // Distanz kleiner Jump Distance
9           füge  $x_i$  zu  $S_j$  hinzu;
10      sonst; // Distanz größer Jump Distance
11          füge  $x_i$  zu  $S_j$  hinzu;
12      j:=j+1;
13      initialisiere  $S_j$ ; // initialisiere neues Segment
14      füge  $x_M$  zu  $S_j$  hinzu;

```

**Rückgabe**

```

15    $\mathbf{S} = \{S_1, S_2, \dots, S_N\}$  // Segmente des Laserscans

```

---

**Abbildung 3.6:** Verwendeter Algorithmus zum Segmentieren des Laserscans.

*Unterschiede der verwendeten Implementierung zu Arras :*

*In der Veröffentlichung von Arras wurde nicht genau beschrieben welches Distanzmaß für die Segmentierung verwendet wird. Vorstellbar wären hier die Euklidische Distanz oder die Distanz der Radialkoordinaten zweier Messpunkte. Beide Varianten wurden für diese Masterarbeit implementiert und getestet. Da die Segmentierung mit der Euklidische Distanz hierbei keine signifikant besseren Ergebnisse erzeugte und zusätzlichen Berechnungsaufwand erfordert, kann die Variante über die Radialkoordinate als die Bessere angesehen werden.*

---



### 3.2.3 Merkmalsgenerierung

Das Problem der Merkmalsgenerierung beschäftigt sich nun damit, wie aus den einzelnen Segmenten Werte berechnet werden können, die ihre Eigenschaften gut Beschreiben. Dabei sollten sich die Werte der Segmenten die von Personen erzeugt werden und solche die von der Umgebung stammen möglichst gut unterscheiden lassen. Arras definierte hierfür folgende 14 Merkmale.

- 1) **Anzahl der Punkte im Segment :**

$$n = |S_i|$$

- 2) **Die Standardabweichung  $\sigma$  :**

$$\sigma = \sqrt{\frac{1}{n-1} \sum_j ||x_j - \check{x}||^2}$$

wobei  $\check{x}$  das Zentrum des Segments bezeichnet.

- 3) **Die durchschnittliche Abweichung  $\varsigma$  vom Median  $\check{x}$  :**

$$\varsigma = \frac{1}{n} \sum_j ||x_j - \check{x}|| \quad \check{x} = \begin{cases} x_{(K+1)/2}, & \text{falls K ungerade} \\ \frac{1}{2}(x_{K/2} + x_{K/2+1}), & \text{falls K gerade} \end{cases}$$

- 4) **Die Jump Distance zum vorherigen Segment :** Diese entspricht der Euklidischen Distanz zwischen dem ersten Punkt des Segments  $S_i$  und dem letzten Punkt des Segments  $S_{i-1}$
- 5) **Die Jump Distance zum nächsten Segment :** Diese entspricht der Euklidischen Distanz zwischen dem letzten Punkt des Segments  $S_i$  und dem ersten Punkt des Segments  $S_{i+1}$ .
- 6) **Breite :** Diese entspricht der Euklidischen Distanz des ersten Punktes des Segments zum letzten Punkt des Segments.
-

- 7) **Die Liniarität** : Dieses Merkmal misst die Linienähnlichkeit eines Segments und entspricht den quadrierten residuen des linear least square Problems. Hierzu verwendeten sie die Hessian  $(\alpha, r)$ -Repräsentation der Ausgleichsgeraden. Sind diese Parameter ermittelt, können durch eq. 3.1 die orthogonalen Abstände der Punkte zur Linie berechnet werden.

$$s_l = \sum_j (x_j \cos(\alpha) + y_j \sin(\alpha) - r)^2 \quad (3.1)$$

- 8) **Kreisähnlichkeit** : Dieses Merkmal entspricht den quadrierten Residuen der Messpunkte zu einem Kreis der in das Segment gelegt wird. Der beste Weg einen solchen Kreis im Sinne der Methode der kleinsten Fehlerquadrate zu finden, ist das Problem mit einem Vektor von unbekannten zu parametrisieren  $x = (x_c \ y_c \ x_c^2 + y_c^2 - r_c^2)^T$ . Wobei  $x_c$  und  $y_c$  den Mittelpunkt des Kreises darstellen und  $r_c$  den Radius. Diese Parameter können gefunden werden durch lösen des überbestimmten Gleichungssystems  $Ax = b$ .

$$A = \begin{pmatrix} -2x_1 & -2y_1 & 1 \\ -2x_2 & -2y_2 & 1 \\ \vdots & \vdots & \vdots \\ -2x_n & -2y_n & 1 \end{pmatrix} \quad b = \begin{pmatrix} -x_1^2 - y_1^2 \\ -x_2^2 - y_2^2 \\ \vdots \\ -x_n^2 - y_n^2 \end{pmatrix}$$

Die Summe der quadrierten Residuen  $s_c$  kann nun berechnet werden durch :

$$s_c = \sum_{i=1}^n (r_c - \sqrt{(x_c - x_i)^2 + (y_c - y_i)^2})^2$$

- 9) **Radius** : Dieses Merkmal entspricht dem Radius  $r_c$  des Kreises aus dem zuvor berechneten Merkmal.
- 10) **Die Länge** : Diese entspricht der Summe der Euklidischen Distanzen zwischen aufeinanderfolgenden Punkten eines Segments  $d_{j,j-1}$

$$l = \sum_j d_{j,j-1}$$


---

- 11) **Die Regelmäßigkeit des Segments** : Diese entspricht der Standardabweichung der Euklidischen Distanzen zwischen zwei aufeinanderfolgenden Punkten des Segments  $d_{j,j-1}$ , zur durchschnittlichen Euklidischen Distanz zwischen den Punkten des Segments  $\check{d}$ .

$$\sigma = \sqrt{\frac{1}{n-1} \sum_j \|d_{j,j-1} - \check{d}\|^2}$$

- 12) **Die mittlere Krümmung** : Die mittlere Krümmung  $\check{k}$  eines Segments, kann mittels einer Approximation berechnet werden. Hierzu sind drei aufeinanderfolgende Punkte  $x_{j-1}, x_j$  und  $x_{j+1}$  gegeben. Die drei Längen  $d_{j-1,j}, d_{j,j+1}$  und  $d_{j-1,j+1}$  stellen dabei die Euklidischen Distanzen zwischen den Messpunkten dar.  $A_j$  bezeichnet die Fläche des Dreiecks, das durch die drei Punkte aufgespannt wird.

$$\check{k} = \sum_j \frac{4A_j}{d_{j-1,j} \cdot d_{j,j+1} \cdot d_{j-1,j+1}}$$

- 13) **Die durchschnittliche Winkeldifferenz** : Dieses Merkmal misst die Konvexität bzw. die Konkavität eines Segmentes. Hierzu wird der Winkel  $\beta_j$  zwischen den beiden Vektoren drei adjazenter Punkte  $\overline{x_{j-1}, x_j}$  und  $\overline{x_j, x_{j+1}}$  berechnet. Anschließend wird ein gemittelter Wert über alle so berechneten Winkel für das Segment ermittelt.

- 14) **Die mittlere Geschwindigkeit** : Die mittlere Geschwindigkeit  $v_j$  eines Segmentpunkts  $p_j$  entspricht der Änderung seiner Radialkoordinate in einem Zeitintervall  $[T_k, T_{k+1}]$ .

$$v_j = \frac{r_j^{k+1} - r_j^k}{T_j^{k+1} - T_j^k}$$

Für das Merkmal wird der Mittelwert über alle  $v_j$  eines Segments gebildet.

*Unterschiede der verwendeten Implementierung zu Arras :*

---

*Das letzte Merkmal 14, beschreibt die Eigenschaft von Personen, als sich bewegende Segmente im Laserscan in Erscheinung zu treten. Allerdings wurde es schon von Arras nur in einem einzigen Test verwendet. Dieser zeigte, dass es nur zu einer marginalen Verbesserung beiträgt und somit Personen auch ohne jegliche Bewegungsmerkmale erkannt werden können. Deshalb ist dieses Merkmal in der verwendeten Implementierung überhaupt nicht vorhanden.*

### 3.2.4 Klassifikation

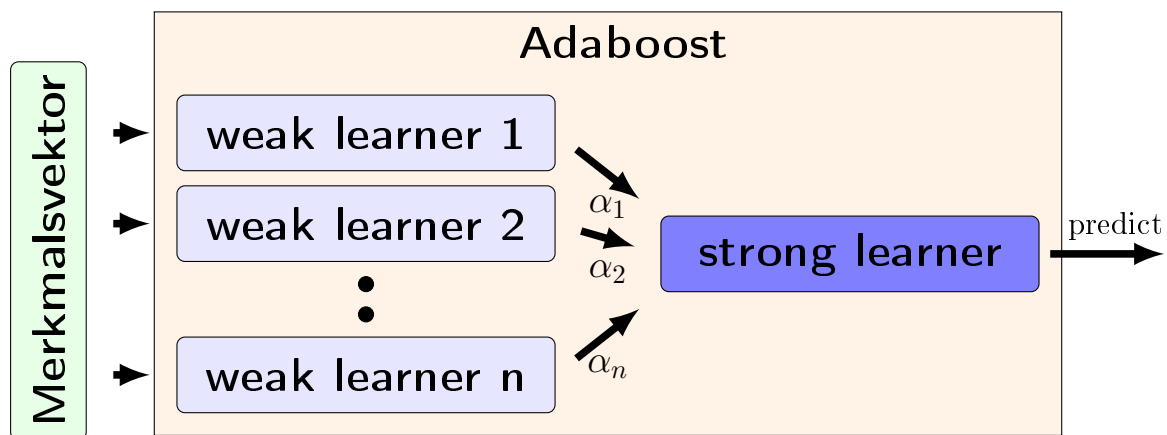
Der Klassifikationsschritt beschäftigt sich nun mit der Frage, wie aus den reelwertigen Merkmalsvektoren die im vorhergehenden Schritt aus den Segmenten berechnet wurden, eine Entscheidung getroffen werden kann ob diese Segmente von einer Person erzeugt wurden oder von der Umgebung stammen. Besonders geeignet für eine solche Entscheidungsfindung haben sich maschinelle Lernverfahren erwiesen. In [PREMEBIDA et al., 2009] kann ein Vergleich gefunden werden, zwischen den Verfahren Gaussian Mixture Model Classifier, Support Vector Machine, Lineare Diskriminanzanalyse, Neuronale Netzwerken und Bayes Filtern. Wobei die Verwendeten Merkmale den zuvor vorgestellten sehr ähnlich sind. Arras sowie die für diese Arbeit verwendete Implementierung nutzen für die Klassifikation hingegen das Adaboost Verfahren

#### Das Adaboost Verfahren

Das Adaboost Verfahren (kurz für Adaptive Boosting), ist ein mit dem Gödel Preis ausgezeichnete meta Algorithmus. Er kann mit vielen anderen maschinellen Lernverfahren kombiniert werden um ihre Performanz zu verbessern. Boosting steht dabei für die Methode mehrere einfache Klassifikatoren (Weak Learner) zu einem besseren Klassifikator (Strong Learner) zusammenzufügen (siehe Abbildung 3.7). Mittels einer gewichteten Mehrheitsentscheidung ist es dann möglich weitaus bessere Ergebnisse zu erzielen, als es mit den einzelnen Klassifikatoren möglich wäre. Die verwendeten Weak Learner können dabei selber eine hohe Fehlerquote aufweisen, solange sie aber leicht besser als Raten sind, können sie den Strong Learner verbessern. Adaptiv steht für das Paradigma von Adaboost, das später trainierte Weak dahingehend optimiert wer-

---

den, sodass sie versuchen besonders die Trainingsdaten richtig zu klassifizieren, die bis dahin noch falsch klassifiziert werden. Adaboost ist dabei weit weniger anfällig für das Problem der Überanpassung an einen bestimmten Trainingsdatensatz, als andere maschinelle Lernverfahren. Allerdings wurde von Phillip Long und Rocco A. Servedio in [LONG und SERVEDIO] kritisiert, dass es in manchen Fällen sehr stark auf Rauschen und Ausreißer reagiert.



**Abbildung 3.7:** Schema des Adaboost Verfahrens.

*Jeder Weak Learner übernimmt den Merkmalsvektor und liefert eine Entscheidung über die Klassenzugehörigkeit des Samples. Diese Entscheidungen werden mittels Wichtungsfaktoren  $(\alpha_1 \cdots \alpha_n)$  vom Strong Learner zur endgültigen Entscheidung fusioniert.*

### Varianten von Adaboost

Durch die Stärke des Adaboost Verfahrens wurden über die Jahre mehrere Abwandlungen entwickelt. Eine Übersicht kann in [FERREIRA, 2007] gefunden werden. Arras verwendete eine Variante, welches unter dem Namen Discrete-Adaboost bekannt ist. Auch bei dieser Masterarbeit stellte sich diese Variante als eine der besten heraus, um das Problem der Personendetektion zu behandeln. Eine vergleichende Evaluation der Varianten Discrete-, Real-, Logit-, und Gentle-Adaboost kann in Anhang A.1 gefunden werden.

### Weak Learner

Die von Arras verwendeten Weak Learner entsprechen denen, die schon Viola und Jones in [VIOLA und JONES, 2001] vorgestellt hatten (siehe eq. 3.2). Ein Weak Learner  $h_t$  verwendet hierbei nur ein einzelnes Merkmal  $f_t$  und einen

$$h_t(x) = \begin{cases} +1 & \text{falls } f_t(x) < \theta_t \\ -1 & \text{sonst} \end{cases} \quad (3.2)$$

$$H(x) = \text{sign} \left( \sum_{t=1}^T \alpha_t h_t(x) \right) \quad (3.3)$$

Threshold  $\theta$  um die Klassenzugehörigkeit zu bestimmen. Der Strong Learner eq. 3.3 bestimmt mit den Entscheidungen der Weak Learner und ihrem Wichtungsfaktor  $\alpha_t$  die endgültige Klassenentscheidung.

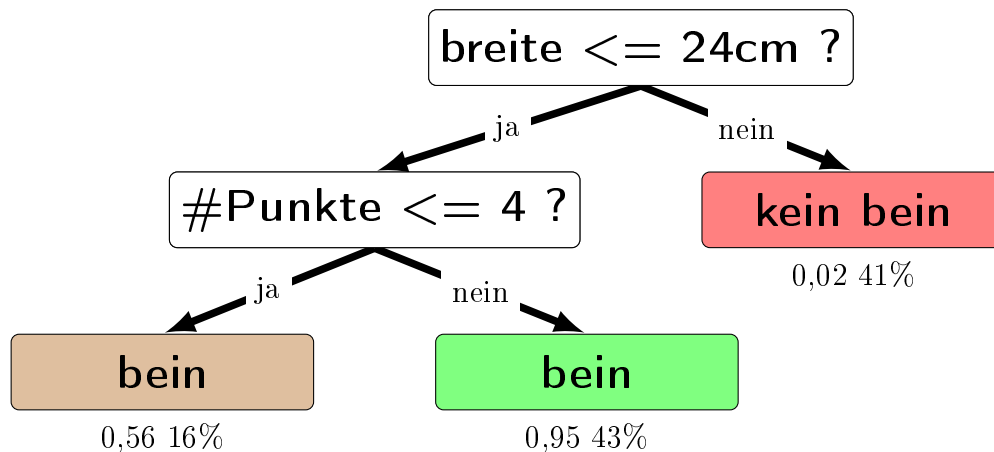
*Unterschiede der verwendeten Implementierung zu Arras :*

*Im Gegensatz zu Arras kommen in der vorhandenen Implementierung Binary Decision Tree Classifier als Weak Learner zum Einsatz. Diese zeigten in der Evaluierungsphase allgemein bessere Ergebnisse. Der Vergleich kann in 5.4 gefunden werden.*

Decision Trees sind gerichtete Bäume, die anhand der Merkmale hierarchisch Entscheidungen treffen. Sie bestehen immer aus einem Wurzelknoten, beliebig vielen inneren Knoten und mindestens zwei Blättern. Die Knoten korrespondieren dabei mit einer der Inputvariablen und repräsentieren dabei logische Regeln, anhand derer entschieden wird, wie ein zu klassifizierender Merkmalsvektor weiter zu behandeln ist. Die Blätter des Baums liefern die Antwort auf das Entscheidungsproblem, zu welcher Klasse ein Sample gezählt wird und eine Wahrscheinlichkeitsverteilung wie sicher diese Entscheidung ist. In Abbildung 3.8 ist ein einfacher Decision Tree für das Entscheidungsproblem zu sehen, ob ein Segment durch ein Bein oder Hintergrund erzeugt wurde. Die erste Entscheidung wird über die Breite des Segments getroffen. Für eine Breite größer 24 cm kann eine relativ sichere Entscheidung von 0,02 (Klassenlabel 0) getroffen werden, dass dieses kein Bein ist. Wobei 41% der Trainingsdaten in dieses Blatt vielen. Für kleinere Segmente muss eine weitere Entscheidung anhand der Anzahl der Punkte getroffen werden. Wobei weitere 43% der Trainingsdaten sicher (0.95 Klassenlabel 1) als

Beine identifiziert werden. Für die restlichen 16% kann allerdings keine sichere Entscheidung getroffen werden.

Die Sicherheit der Entscheidung entspricht dabei dem Rückgabewert des Weak Learners, welcher mit den Wichtungsfaktoren von Adaboost multipliziert wird um eine Entscheidung über die Klassenzugehörigkeit zu treffen.



**Abbildung 3.8:** Beispiel eines Decision Trees für die Beindetektion.

*Knoten weiß, Blätter nach Klassenzugehörigkeit eingefärbt. Erstellt mit dem Python scikit-learn Modul.*

Das Training eines Baumes beginnt mit dem Wurzelknoten. Anhand eines bestimmten Splitkriteriums wird eine Variable des Merkmalsvektors und ein Threshold ermittelt, welcher den Trainingsdatensatz am besten in zwei Teile untergliedert. In der Implementierung wurde hierfür der minimale quadrierte Fehler über den Teildatensätzen verwendet. Diese Prozedur wird nun so lange auf den jeweiligen Teildatensätzen fortgeführt, bis entweder alle Daten eines Teilsets die selbe Klasse besitzen oder ein weiterer Split keine zusätzliche Information mehr bereitstellt.

Ein großer Nachteil der Decision Trees ist, dass diese stark dazu neigen sich an einen Trainingsdatensatz überanzupassen. Dabei liefern sie zwar gute Ergebnisse für Daten aus dem Trainingsdatensatz, für andere Daten kann die Klassifikationsgüte aber stark schlechter sein. Eine Möglichkeit diesem entgegenzuwirken wird als Pruning bezeich-

net. Dabei wird mittels Crossvalidation (siehe Abschnitt 3.3) ein Fehler über einen Validierungsdatensatz berechnet, der nicht für das Training verwendet wurde. Nun werden nach und nach Knoten aus dem Baum entfernt, bis dieser Fehler minimal wird.

### 3.2.5 Generierung der Personenhypothese

Das Problem der Generierung der Personenhypothese beschäftigt sich nun damit, wie und wo aus den klassifizierte Segmenten ein Ort generiert werden kann, an dem eine Person angenommen wird. Arras behandelte dieses Problem überhaupt nicht und verwies dabei darauf, dass es sich hierbei um ein komplexes Datenassoziationsproblem handelt, und somit besser von einer höheren Ebene behandelt werden kann. Der am Fachgebiet eingesetzte Tracker arbeitet jedoch auf bereits bestehenden Personenhypothesen und fusioniert diese aus unterschiedlichen Sensoren. Deshalb kommt eine Heuristik zum Einsatz, welche aus den Orten positiv klassifizierter Segmente eine Hypothese generiert. Der bisherige Ansatz hierbei war, nur dann eine Person anzunehmen wenn zwei Beine in einem bestimmten Radius erkannt wurden, der dem maximal möglichen Beinabstand einer Person entspricht. Durch Experimente konnte gezeigt werden, dass weit bessere Ergebnisse erzielt werden können, wenn bereits ein erkanntes Bein für eine Personenhypothese ausreicht (siehe Abschnitt A.2). Der neu implementierte Algorithmus zur Hypothesengenerierung kann in 3.9 gefunden werden. Dieser übernimmt die Orte aller positiv klassifizierten Segmente und fusioniert diese, falls sie sich in einem bestimmten Radius befinden. Falls kein korrespondierendes Segment gefunden wurde, wird der Ort des einzelnen Segments als Hypothese angenommen. Um nicht die Eigenschaft zu verlieren, dass zwei Beine grundsätzlich eine stärkere Hypothese generieren, wird zusätzlich eine Confidence berechnet welche die Sicherheit der Hypothese angibt. Diese ist der normierte reelwertige Rückgabewert des Adaboost Strong Learners. Falls eine Hypothese aus zwei Segmenten generiert wurde, wird die Confidence dieser beiden addiert.

---



**Eingaben**

- 1      $S^p = \{s_1^p, s_2^p, \dots, s_N^p\}$      // Kandidatenmenge für Hypothesen  
 2      $maxLegDist$      // maximal mögliche Abstand von Beinen

**Algorithmus**

- 3      $H = \{\}$ ;     // initialisiere die Hypothesenmenge  
 4     Für jeden  $s_i^p \in S^p$ ;  
 5         finde ein  $s_j^p \in S^p : j \neq i, dist(s_i^p, s_j^p) = \text{minimal}$ ;  
 6         falls  $dist(s_i^p, s_j^p) < maxLegDist$ ;  
 7             berechne Mittelpunkt  $m$  zwischen  $s_i^p$  und  $s_j^p$ ;  
 8             füge  $m$  in  $H$  ein;  
 9              $S^p := S^p \setminus \{s_i^p \cup s_j^p\}$ ;     // entferne segmente aus Kandidatenmenge  
 10         sonst;     // kein weiteres Bein im möglichen Radius  
 11             füge Zentrum von  $s_i^p$  in  $H$  ein;  
 12              $S^p := S^p \setminus s_i^p$ ;     // entferne Segment aus Kandidatenmenge

**Rückgabe**

- 13      $H = \{H_1, H_2, \dots, H_M\}$      // Hypothesen

---

**Abbildung 3.9:** verwendeter Algorithmus zum generieren der Personenhypothese

---

### 3.2.6 Generierung der Trainingsdaten

Arras verwendete für seine Experimente Trainingsdaten, welche manuell gelabelt wurden. Für einen Einsatz in schwierigeren Umgebungen sind allerdings größere Datensätze notwendig, welche nicht mehr manuell erstellbar sind. Deshalb wurde hierfür ein Verfahren verwendet welches zuerst ein Hintergrundmodell einer Umgebung lernt, welche keine Personen enthält. Anhand dieses Modells kann der Datensatz nun automatisiert nach Hintergrund und Vordergrund gelabelt werden, welcher durch Personen erzeugt wurde. Zusammen mit dem eingangs beschriebenen Methode zur Klassifizierung von Erscheinungsformen, kann gezielt ein positiver Trainingsdatensatz erstellt werden. Der Pseudocode für die automatische Klassifizierung der Erscheinungsformen

---

kann in Anhang C.1 gefunden werden. Für Negative Trainingsdaten kann der Roboter durch Umgebungen gefahren werden, die keine Personen beinhalten.

### 3.3 Crossvalidation und Learning Curves

Bei Maschinellen Lernverfahren ist in der Praxis die Qualität der Klassifikation von mehreren Einflussfaktoren Abhängig. So können die Größe des Trainingsdatensatzes, die Anzahl der verwendeten Merkmale oder die Parameter des Klassifikators die Resultate stark beeinflussen. Ein trainierter Klassifikator kann dabei unter zwei unterschiedlichen Problemen leiden, die als Über- bzw. Unteranpassung bezeichnet werden. In der Fachsprache auch hoher Bias bzw. hohe Varianz genannt.

Das Problem der Überanpassung bzw. hohe Varianz besteht darin, dass ein Klassifikator zu stark an den verwendeten Trainingsdatensatz angepasst ist. In diesem Fall werden für die Trainingsdaten sehr gute Ergebnisse geliefert. Für Daten die sich nicht in diesem Datensatz befinden können die Ergebnisse jedoch weitaus schlechter sein. Er ist also nicht in der Lage zu generalisieren, was sich auf den praktischen Einsatz denkbar nachteilig auswirkt.

Im Gegensatz dazu, kann der Klassifikator aber auch Unterangepasst sein bzw. einen hohen Bias besitzen. In diesem Fall werden sowohl für Daten aus dem Trainingsdatensatz, als auch für Daten die sich nicht in diesem befinden schlechte Ergebnisse erzielt, was sich für den praktischen Einsatz ebenso nachteilig auswirkt.

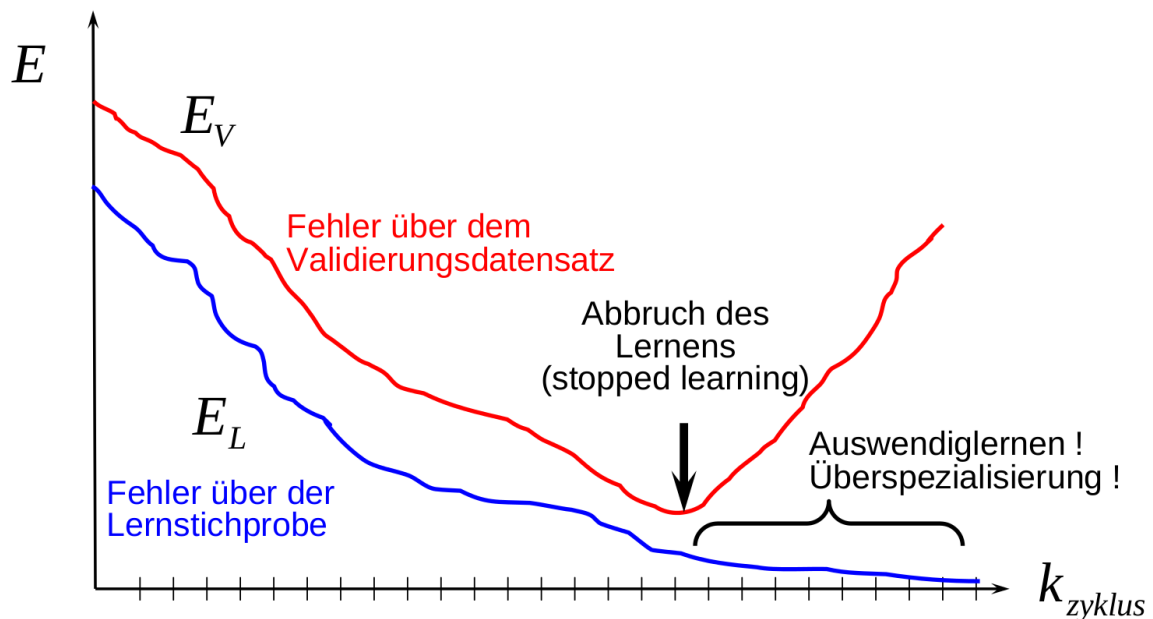
Man ist also versucht einen Klassifikator dahingehend zu optimieren, dass er sowohl eine kleinen Fehler über den Trainingsdatensatz erzeugt als auch eine möglichst gute Generalisierungsfähigkeit besitzt. Leider ist nicht immer klar ersichtlich welche Handlungen die Ergebnisse verbessern oder verschlechtern würden. So kann die Hinzunahmen neuer Trainingsamples einen Klassifikator mit hoher Varianz verbessern. Bei Klassifikatoren mit hohem Bias bewirken diese neuen Samples allerdings keine Änderung. Um die Probleme eines trainierten Klassifikators zu analysieren und die Optimierung zu erleichtern stehen zwei Verfahren zur Verfügung, die Crossvalidation und die Learning Curves.

---

Um dem Problem der hohen Varianz zu begegnen, wird in der Praxis die Technik der Crossvalidierung eingesetzt (siehe Abbildung 3.10). Dazu wird der vorhandene Datensatz in drei Teile aufgespalten die unterschiedliche Sample beinhalten. Den Trainingsdatensatz, Crossvalidierungsdatensatz und den Testdatensatz. Dabei sollte der Trainingsdatensatz ungefähr 50% der vorhandenen Sample beinhalten. Der Crossvalidierungs- und Testdatensatz sollten jeweils 25% der Sample enthalten. Mit dem Trainingsdatensatz wird der Klassifikator trainiert und ein Fehler über die Trainingsschritte ermittelt. Der Validierungsdatensatz fließt nicht in das Training ein, sondern wird dazu verwendet Generalisierungseigenschaften über Daten zu ermitteln mit denen nicht trainiert wurde. Auf diesen Datensatz wird ebenso für jeden Trainingsschritt ein Fehler berechnet. Beginnt dieser an einem bestimmten Punkt wieder zu steigen, der Klassifikator sich also überanzupassen, wird das Training abgebrochen. Es wird dann der Klassifikator verwendet, der das lokale globale Minimum des Validierungsfehlers erzeugte. Der Testdatensatz besteht aus Daten die überhaupt nicht in das Training einfließen und wird zur Bestätigung der erzielten Ergebnisse verwendet.

Im Gegensatz zur Crossvalidation, die nur die Überanpassung eines Klassifikators verhindert, kann mit der Methode der Learning Curves genauer analysiert werden welche Probleme ein trainierter Klassifikator besitzt. Dies erleichtert es Entscheidungen zu Treffen mit welchen Parametern ein neues Training gestartet werden soll. Hierzu werden die Daten wieder in einen Trainings- und einen Validierungsdatensatz unterteilt. Im Gegensatz zur Crossvalidation, besitzen diese aber keine feste Größe. Es wird damit begonnen einen Klassifikator mit einen relativ kleinen Trainingsdatensatz zu trainieren. Der Rest der Samples wird für die Validierung verwendet. Nun werden nach und nach Klassifikatoren trainiert, die immer größere Trainingsdatensätze und somit kleinere Validierungsdatensätze verwenden. Für jeden dieser Klassifikatoren wird der Fehler über die beiden Datensätze berechnet und in zwei kurven geplottet (siehe Abbildung 3.11). Mit zunehmender Größe des Trainingsdatensatzes beginnen sich die beiden Kurven anzunähern. Im Bild links ist ein Klassifikator mit hohem Bias zu sehen. Dies ist an einem relativ großen Fehler

---



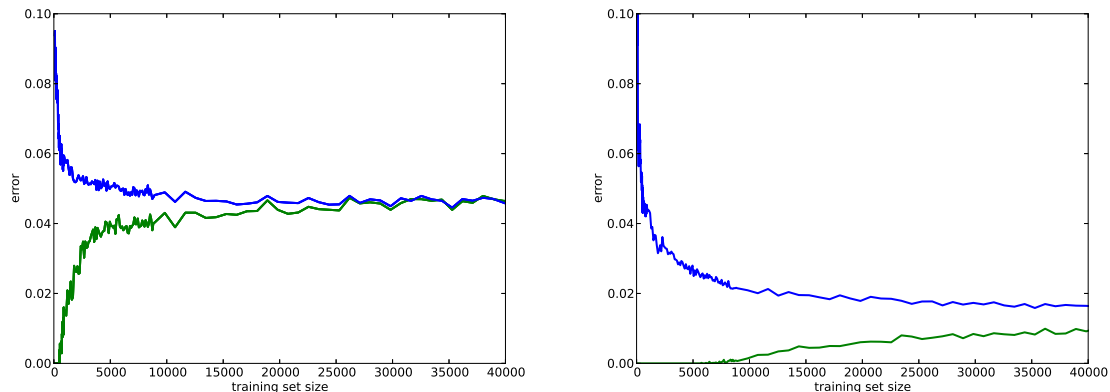
**Abbildung 3.10:** Konzept der Crossvalidation.

Der Fehler  $E_L$  des Trainingsdatensatzes sinkt kontinuierlich mit zunehmender Trainingsdauer. Der Fehler  $E_V$  über den Validierungsdatensatz beginnt ab einem bestimmten Zeitpunkt wieder zu steigen, an dem der Klassifikator sich Überanpasst. (Abbildung übernommen aus der "Angewandte Neuroinformatik" Vorlesung des FG NIKR).

über den beiden Datensätzen zu erkennen. Eine Hinzunahme neuer Trainingssamples würde hier keine Verbesserung bewirken, da die Beiden Kurven bereits den selben Fehler aufweisen und sich dieser mit größeren Trainingsdatensätzen nicht ändern würde. Rechts im Bild Hingegen ist ein Klassifikator mit hoher Varianz zu sehen. Dies kann erkannt werden da die beiden Kurven einen noch relativ großen Unterschied bei den erzeugten Fehlern aufweisen. Mit der Hinzunahme weiterer Trainingsdaten würde der Fehler über den Trainingsdatensatz weiter steigen und der Fehler über den Validierungsdatensatz weiter sinken, was die allgemeine Klassifikationsgüte verbessern würde.

### weitere Optimierungsmöglichkeiten

Neben der Anpassung der Größe des Trainingsdatensatzes stehen noch weitere Mög-



**Abbildung 3.11:** Learning Curve eines Klassifikators.

***Links :** mit hohem Bias. **Rechts :** mit hoher Varianz. Fehler über den Trainingsdatensatz grün. Fehler über den Validierungsdatensatz blau. Erstellt mit dem beschriebenen Adaboost Klassifikator und einer Baumtiefe von eins (links) und drei (rechts). Trainiert mit einer Trainingsdatensatzgröße von bis zu 40000 Samples.*

lichkeiten zur Verfügung die Probleme der Über- und Unteranpassung zu behandeln.

**Anzahl verwendeter Merkmale :** Für einen Klassifikator mit hohem Bias kann die Hinzunahme weiterer Merkmale die Klassifikationsgüte erheblich verbessern. Damit stehen dem Klassifikator mehr Informationen zur Verfügung, anhand derer er seine Entscheidung treffen kann. Dadurch kann eine Verringerung des Fehlers über sowohl Trainings- als auch Testdaten erzeugt werden. Für Klassifikatoren mit hoher Varianz kann es nützlich sein die Anzahl der verwendeten Merkmale zu verringern. Hierzu existieren Merkmalsselektionsalgorithmen zur Verfügung, welche die Merkmale mit dem höchsten Informationsgehalt ermitteln können.

**Hinzunahme von Komplexität :** Wie in Abbildung 3.11 konnte die Erhöhung der Baumtiefe der Weak Classifier den Bias verringern und somit den Fehler über beide Datensätze verringern. Die meisten maschinellen Lernverfahren besitzen Möglichkeiten die Komplexität zu erhöhen. Weshalb dies auch allgemein eine weitere Möglichkeit ist einem hohen Bias zu begegnen.

**Regularisierung :** Regularisierung ist eine weitere Methode um die Überanpassung bei den meisten maschinellen Lernverfahren zu verhindern. Hierzu wird ein Strafterm an algorithmusspezifische Trainingsparameter angehängt. Durch Crossvalidierung kann hiermit ein lokales Minimum im Fehlergebirge des Validierungsdatensatzes gefunden werden.

*Anmerkung : Für unbalancierte Datensätze ist bei den Learning Curves zu beachten, dass positive und negative Samples immer im gleichen Verhältnis in den Trainings- und Validierungsdaten enthalten seien sollten. Für ungleiche Verteilungen zeigten sich stark abweichende Kurvenformen, was die Analyse erschweren kann.*



## Kapitel 4

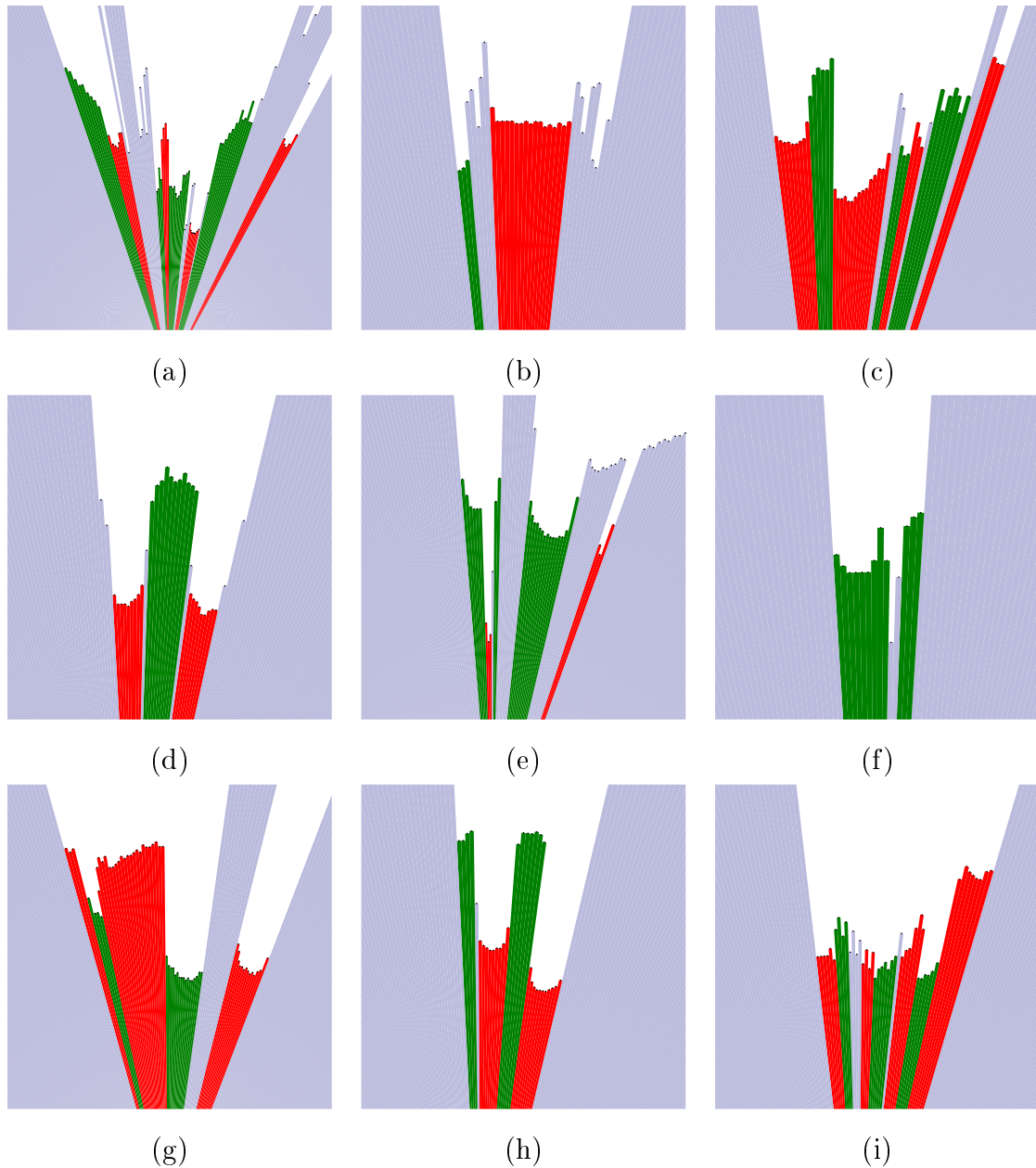
### Eigener Ansatz

Dieses Kapitel beschäftigt sich mit einem neuen Ansatz für die Personendetektion in Laserscans auf mobilen Robotersystemen. Die Motivation für dieses Verfahren bestand darin, dass Personen im Laserscan in unterschiedlichen Erscheinungsformen vorkommen können. Wie im vorherigen Kapitel beschrieben, können die durch Beine erzeugte Segmente auch gemerged und verdeckt vorkommen, weshalb die auf Kreisähnlichkeit ausgelegten Merkmale von Arras ungeeignet erschienen. In speziellen Einsatzumgebungen wie Krankenhäusern oder Altenheimen kommen zu diesen Erscheinungsformen noch weitere hinzu. So können Krücken Beinsegmente zerteilen oder mit ihnen ein Segment bilden und dadurch ebenso ihr Aussehen stark verändern (siehe Abbildung 4.1 (e) und (f)). Rollatoren können Beinsegmente sogar teilweise ganz verdecken, wobei sich die durch Rollatoren erzeugte Segmente nicht stark von kurzen Wandstücken unterscheiden (siehe Abbildung 4.1 (g) bis (i)). Betrachtet man Personen in Rollstühlen (siehe Abbildung 4.1 (a) bis (d)), ist es nahezu unmöglich geeignete mathematische Modelle zu finden, welche diese Erscheinungsformen gut beschreiben.

Deshalb wurden neue Merkmale Implementiert, die nicht mehr geometrischen und statistischen Eigenschaften der Segmente entsprechen, sondern direkt aus den Tiefeninformationen der Lasermessungen generiert werden. Ziel ist es dabei nicht mehr nur einzelne Segmente zu Klassifizieren, sondern für alle Messpunkte eines Objektes Entscheidungen zu treffen. Für die Problemstellung dieser Masterarbeit bedeutet

---





**Abbildung 4.1:** Darstellung von Rollstühlen, Rollatoren und Personen mit Krücken  
*(a)* Rollstuhl in seitlicher Rückansicht. *(b)* Rollstuhl von hinten. *(c)* Rollstuhl in seitlicher Frontalansicht. *(d)* Rollstuhl Frontal. *(e)* Kruecken zerteilen Beine und bilden Segmente. *(f)* Kruecke schneidet Beine und bildet kein Segment. *(g)* Rollator mit zwei Beinen rechts. *(h)* Beine verdecken Rollator. *(i)* Rollator verdeckt Beine. Segmente mit mehr als 3 Punkten abwechselnd rot und grün markiert.

das, die Klassifikation von Beinpaaren, Rollstühlen und Rollatoren vorzunehmen. Der erste Teil dieses Kapitels beschäftigt sich mit der grundlegenden Idee, welche hinter diesen neuen Merkmalen steckt. Der zweite Teil, beschreibt den Ablauf, wie eine Personendetektion mit diesen neuen Merkmalen vonstatten geht.

## 4.1 Die Grundidee

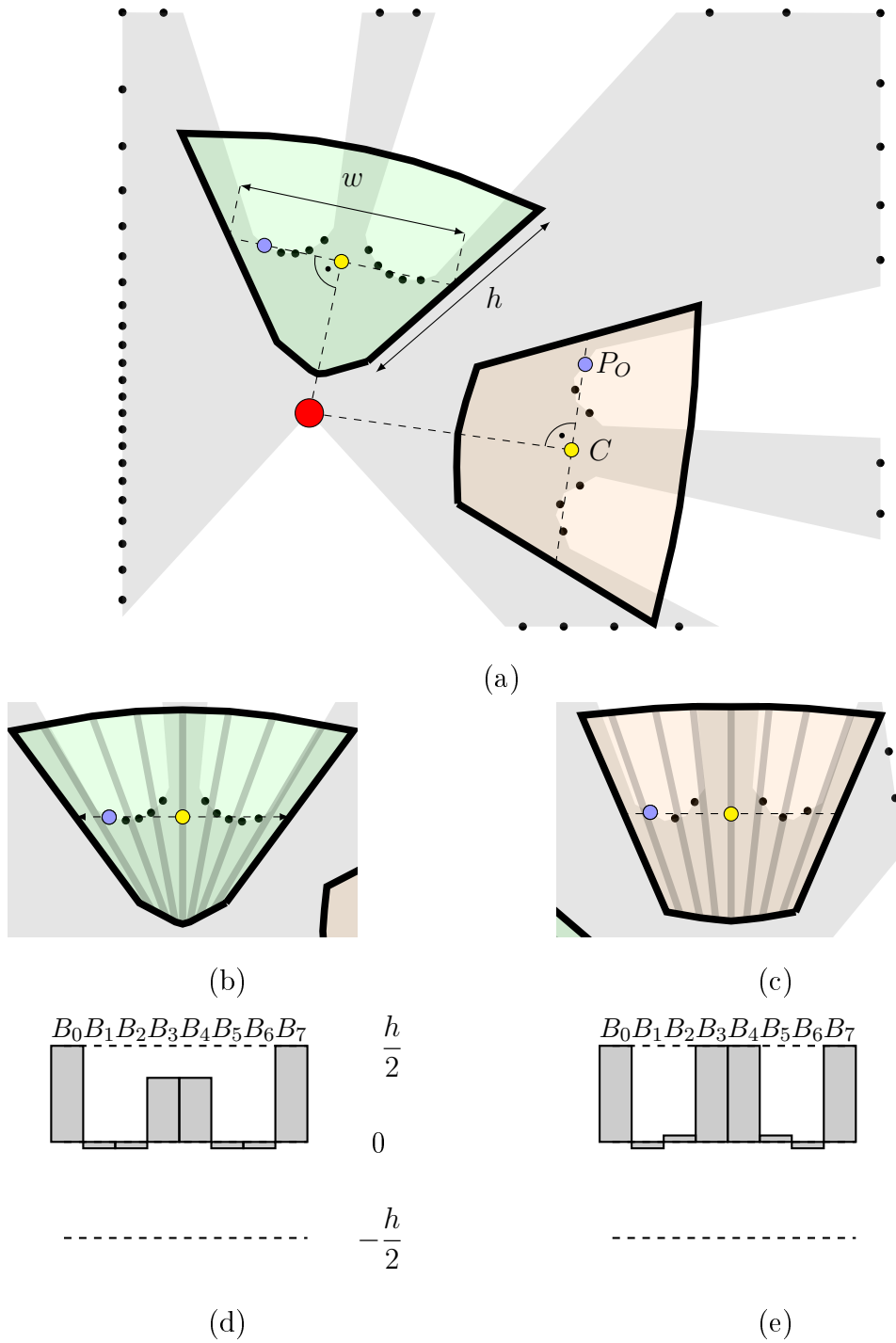
Die Grundidee des Verfahrens besteht darin, benachbarte Entfernungsmessungen des Laserscans in einen Raum zu projizieren, in dem das selbe Objekt unabhängig von Entfernung und Messwinkel immer ein gleiches Aussehen besitzt. Sie wurden deshalb GDIF genannt, kurz für **G**eneric **D**istanc **I**nvariant **F**eatures. Hierzu werden lokale Bereiche des Laserscans “ausgeschnitten” und in ein lokales Koordinatensystem transformiert. Um dies zu bewerkstelligen, werden Bounding Boxen in den Scan gelegt, sodass diese das komplette Objekt umfassen (siehe Abbildung 4.2 (a)).

Um den radialen Eigenschaften eines Laserscans dabei gerecht zu werden, sind diese nicht quadratisch, sondern werden nur durch ihren Mittelpunkt  $C$ , ihre Mittellinie der Breite  $w$  und ihre Höhe  $h$  definiert. Die Parameter  $w$  und  $h$  werden dabei durch die Größe des zu klassifizierenden Objekts bestimmt und müssen spezifisch für dieses ermittelt werden.  $C$  wird dabei aus einem Ursprungspunkt  $P_O$  berechnet, welcher immer ein Scanpunkt eines Objektes ist und den größten Messwinkel besitzt. Um die Unabhängigkeit des Aussehens vom Winkel im Laserscan zu gewährleisten, wird die Mittellinie so ausgerichtet, sodass sie immer orthogonal zur Radialkoordinate des Mittelpunktes steht.

Der Mittelpunkt der Box stellt nun den Ursprung des neuen lokalen Koordinatensystems dar. Alle Messpunkte, deren Scanlinien einen Schnittpunkt mit der Mittellinie der Box besitzen, werden nun in Relation zu dieser betrachtet um einen Merkmalsvektor zu generieren. Alle Scanpunkte die keinen Schnittpunkt mit der Mittellinie besitzen werden nicht weiter betrachtet.

Da die selben Objekte je nach Entfernung zum Sensor eine unterschiedliche Anzahl an Messpunkten besitzen, müssen die Tiefeninformationen nun in einen Merkmalsraum

---



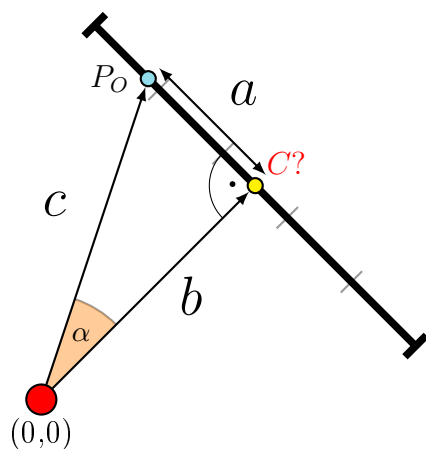
**Abbildung 4.2:** Schematische Darstellung für die Berechnung der GDIF Merkmale  
 (a) der komplette Laserscan mit Bounding Boxen. Das in (d) dargestellte Tiefenhistogramm korrespondiert mit der Bounding Box in (b). Ebenso für (e) und (c). Ursprungspunkt  $P_O$  blau. Mittelpunkt  $C$  gelb. Dargestellt wurden die gemittelten Tiefenwerte von Histogrammen mit 8 Bins

mit statischer Größe umgewandelt werden. Dies wird erreicht, indem die Box in eine bestimmte Anzahl gleichgroßer 'Bins' unterteilt wird (siehe Abbildung 4.2 (b) und (c)). In jedem dieser Bins werden nun Merkmale aus allen Messpunkte ermittelt, welche den Abschnitt des jeweiligen Bins auf der Mittellinie schneiden. Hierzu werden zunächst die Abstände der Messpunkte zum Schnittpunkt mit der Mittellinie berechnet. Falls ein Messpunkt weiter entfernt ist als die definierte Höhe  $h$  der Box, wird der Abstand auf einen Minimal- bzw. Maximalwert gesetzt. Wobei dieser Wert der halben Höhe der Box entspricht, falls der Messpunkt weiter vom Sensor entfernt ist als der Schnittpunkt mit der Mittellinie. Falls sich der Messpunkt näher am Sensor befindet als der Schnittpunkt mit der Mittellinie, wird dieser Wert auf die negative halbe Höhe der Box gesetzt. Geometrisch betrachtet werden also Punkte, die sich hinter oder vor der Box befinden, auf ihre Ränder projiziert, wodurch ihre spezifische Form entsteht. Als endgültige Merkmale werden aus den lokalen Tiefeninformationen der einzelnen Bins jeweils drei Werte berechnet. Diese sind der kleinste gemessene lokale Tiefenwert, der größte gemessene lokale Tiefenwert und der Durchschnitt aller gemessenen lokalen Tiefenwerte. Die sich hieraus ergebenden Tiefenhistogramme (siehe Abbildung 4.2 (d) und (e)) stellen nun den Merkmalsvektor dar, den dieses Verfahren verwendet. Anzumerken ist, dass Boxen, die über die Grenzen der Laserscans hinausragen, verworfen werden müssen. Somit wird der Detektionsbereich im Vergleich zum Arras-Verfahren leicht eingeschränkt. Steht aber ein  $360^\circ$  zur Verfügung, würde dieser Nachteil wegfallen.

### Berechnung der Merkmale

Für die Berechnung des Merkmalsvektors müssen zwei Dinge ermittelt werden. Zum einen die lokalen Tiefenwerte der Messpunkte und zum anderen das Bin, in das diese eingetragen werden. Der erste Schritt hierbei ist die Berechnung des Mittelpunktes  $C$  der Box. Damit sich diese im Zentrum des Objektes befindet, kann die in Abbildung 4.3 dargestellte Methode verwendet werden. Dabei muss ein Offset  $a$  angegeben werden, der angibt, wie weit das Zentrum  $C$  vom Ursprungspunkt  $P_O$  aus nach rechts verschoben werden soll. Dabei sollte darauf geachtet werden, dass sowohl der Linke, als auch der Rechte Rand der Box über das Objekt hinausragt. Damit wird sichergestellt,

---



$$c = r_{P_O}$$

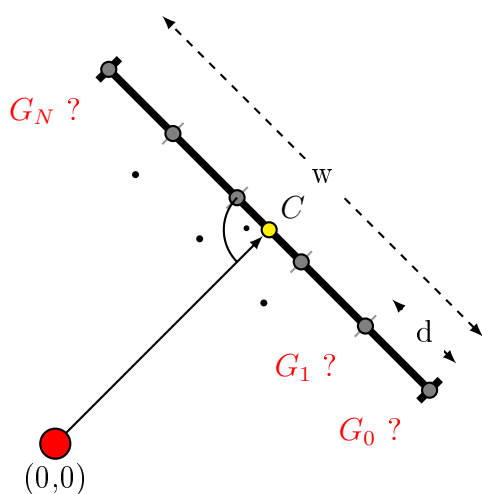
$$\alpha = \arcsin\left(\frac{a}{c}\right)$$

$$r_C = b = \sqrt{c^2 - a^2}$$

$$\phi_C = \phi_{P_O} - \alpha$$

**Abbildung 4.3:** Darstellung zur Berechnung des Zentrums der Box.

Der Offset  $a$  wird so festgelegt, sodass ein gewisser Bereich neben dem Objekt in die Bounding Box hineinfällt.  $c$  ist bestimmt durch die Radialkoordinate des Ursprungspunktes  $P_O$



$$\phi_{ort} = \phi_C + \frac{\pi}{2}$$

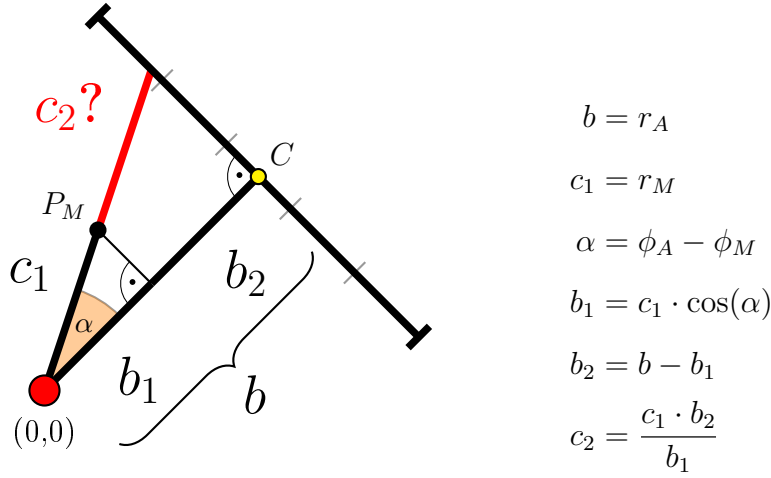
$$d = \frac{w}{N}$$

$$x_{Bi} = x_C + \cos(\phi_{ort}) \cdot \left((d \cdot i) - \frac{w}{2}\right)$$

$$y_{Bi} = y_C + \sin(\phi_{ort}) \cdot \left((d \cdot i) - \frac{w}{2}\right)$$

**Abbildung 4.4:** Darstellung für die Berechnung der Bingrenzen  $G_0, \dots, G_N$ .

Die Punkte der Bingrenzen auf der Mittellinie werden orthogonal zur Winkelkoordinate des Mittelpunktes  $C$  ausgerichtet.  $d$  ist bestimmt durch die Breite  $w$  dividiert durch die Anzahl der Bins.



**Abbildung 4.5:** Trigonometrische Relationen von Messpunkt und Mittellinie der Box für die Berechnung der lokalen Tiefenwerte.

$c_1$  ist bestimmt durch die Radialkoordinate des Messpunktes  $P_M$ , für den der lokale Tiefenwert  $c_2$  berechnet werden soll.  $b$  ist bestimmt durch die Radialkoordinate des Mittelpunktes  $C$ .

dass auch das Minimalitätsmerkmal von Personen im Merkmalsvektor mit abgebildet wird, da hier bei Personen oft Freiflächen enthalten sind.

Um zu ermitteln in welches Bin der lokale Tiefenwert eines Messpunktes eingetragen werden soll, müssen die Punkte  $G_0, \dots, G_N$  bekannt sein, an denen die Grenzen der Bins die Mittelgerade der Box schneiden. Wobei  $N$  durch die Anzahl der Bins bestimmt wird. Die Formeln zur Berechnung können Abbildung 4.4 gefunden werden. Transformiert man die so berechneten kartesischen Koordinaten in Polarkoordinaten, lässt sich durch einfache Vergleiche mit den Scanwinkeln der Messpunkte der Index des Bins ermitteln. Desweiteren müssen nur die Messpunkte des Scans betrachtet werden, die eine Winkelkoordinate größer als  $G_0$  und kleiner als  $G_N$  besitzen. Alle anderen Messpunkte besitzen keinen Schnittpunkt mit der Mittellinie der Box. Ist das Bin bekannt, in den ein lokaler Tiefenwert eingetragen werden soll, muss dieser noch berechnet werden. Die effizienteste Methode hierfür ist in Abbildung 4.5 zu sehen (weitere können in Anhang B gefunden werden). Dabei wird der lokale Tiefenwert mittels trigonometrischer Funktionen und Anwendung des Dreisatzes bestimmt. Sind alle lokalen Tiefenwerte

der einzelnen Bins bekannt, müssen diese nur noch gemittelt werden. Wobei der minimale bzw. maximale Wert pro bin gleich mit bestimmt werden kann. Die so erhaltenen Merkmalsvektoren können dann weiterverarbeitet werden.

## 4.2 Ablauf der Personendetektion

Die Personendetektion mit den GDIF Merkmalen verläuft grundsätzlich in den gleichen aufeinanderfolgenden Schritten, wie die Methode von Arras. Der Vorverarbeitung, Segmentierung, Merkmalsextraktion, Klassifikation und Hypothesengenerierung.

### **Vorverarbeitung und Segmentierung**

Die ersten beiden Schritte der Personendetektion sind gleich mit dem Verfahren von Arras. Anzumerken hierbei ist, dass eine Segmentierung grundsätzlich nicht nötig wäre. Im Gegensatz zu den Merkmalen von Arras, können die GDIF Merkmale auch für Segmente mit weniger als drei Messpunkten berechnet werden. Somit wäre es auch möglich einfach alle Messpunkte als Ursprungspunkte für die Bounding Boxen zu verwenden. Allerdings zeigte sich während der Evaluierung, dass die Klassifizierung von zu kleinen Segmenten die Ergebnisse deutlich verschlechtern würde. Deshalb wurde das Kriterium beibehalten, dass Segmente mindestens drei Punkte besitzen müssen um klassifiziert zu werden. Ein weiterer Vorteil ist, dass der Messpunkt eines Objektes, welcher den größten Messwinkel besitzt, Auch immer der Endpunkt eines Segments ist. Somit können diese Endpunkte direkt als Ursprungspunkte der Box verwendet werden, was zu einem deutlichen Laufzeitvorteil führt.

### **Merkmalsgenerierung und Klassifikation**

Die Merkmalsgenerierung erfolgt wie in Abschnitt 4.1 beschrieben. Für die Klassifikation wurde der vorhandene Klassifikator verwendet, welcher Discrete Adaboost mit Decision Trees als Weak Learner verwendet. Ursprünglich war hierfür vorgesehen mehrere verschiedene Klassifikatoren zu testen. Da aber durch die Optimierung der Trainingsparameter schon sehr gute Ergebnisse erzeugt werden konnten, wurde dieser Schritt weggelassen.

---

### Hypothesengenerierung

Für die Generierung der Personenhypothese wurden zwei Methoden implementiert. Da Personen die durch ihre Beine im Laserscan in Erscheinung treten immer nur ein oder zwei Segmente erzeugen, kann für diese die gleiche Heuristik verwendet werden, welche in Abschnitt 3.2.5 vorgestellt wurde. Die Breite  $w$  der Bounding Box wird dabei auf die maximale Breite von einzelnen Beinsegmenten gesetzt. Für die Klassifikation von Rollstühlen und Rollatoren ist diese Methode allerdings ungeeignet. Diese können vor allem auf nahe Entfernung deutlich mehr Segmente erzeugen, was es schwierig macht eine geeignete Heuristik zu finden. Deshalb würde für diese eine Methode verwendet, bei der die Segmente iterativ mit absteigendem Scanwinkel betrachtet werden. Wird dabei ein Objekt positiv klassifiziert, wird der Mittelpunkt  $C$  der Bounding Box als Ort der Personenhypothese verwendet. Damit ein Objekt nicht mehrere Hypothesen erzeugt, werden alle Segmente in einem bestimmten Umkreis von weiteren Klassifikationen ausgeschlossen. Als Radius dieses Umkreises wurde dabei die Breite der Bounding Box verwendet.

### Generierung von Trainingsdaten

Für die Generierung der positiven Trainingsdaten wird zuerst ein Hintergrundmodell gelernt. Dieses dient zur Unterscheidung von Umgebungssegmenten und Segmenten die von Personen erzeugt wurden. Zusätzlich darf sich in dem Datensatz entweder nur eine Person befinden oder es muss ein minimaler Abstand zwischen Personen definiert werden, welcher niemals unterschritten wird. Die Segmente werden nun sukzessiv mit absteigenden Scanwinkel betrachtet. Für das erste Segment, welches durch eine Person generiert wurde, wird nun eine Bounding Box erzeugt. Für sie werden die Merkmale berechnet und in den positiven Trainingsdatensatz eingefügt. Alle weiteren Segmente innerhalb des minimalen Personenabstandes werden für den weiteren Generierungsverlauf übersprungen. Falls nur Beine detektiert werden sollen, sind diese Einschränkungen nicht notwendig.

Negative Trainingsdaten können einfach erzeugt werden, indem Datensätze verwendet werden in denen keine Personen enthalten sind. Dabei kann die Box an den Endpunk-

---



ten aller Segmenten angesetzt werden.

# Kapitel 5

## Evaluation

### 5.1 Verwendete Lasersensoren

Für das Training und die Evaluation der Verfahren, wurde mit den Laserscannern der mobilen Roboterplattformen SCITOS A5 (Laserhöhe : 40cm) und SCITOS G3 (Laserhöhe : 23cm) Daten aufgenommen. Beide Sensoren verfügen über eine Auflösung von  $0.5^\circ$ , bei einem Scanwinkel von  $270^\circ$ . Somit sind in jedem Laserscan genau 541 Entfernungsmessungen enthalten.

Es wurden dabei nur Messpunkte mit einem Abstand von bis zu 10 Meter vom Sensor betrachtet, was als maximaler Abstand ermittelt wurde, bei denen Personen noch Segmente mit mindestens drei Messpunkten erzeugen.

### 5.2 Datenbasis

#### Positivdaten

Die Positivdatensätze wurden am FG NIKR mit Mitarbeitern des Fachgebietes und in der m&i-Fachklinik Bad Liebenstein hauptsächlich mit Mitarbeitern des Fachgebietes und einem Patienten aufgenommen. Alle Personen trugen dabei entweder lange Jeans oder Trainingshosen.

Für die Evaluierung nach Erscheinungsformen von Personen, wurden die positiven Ground Truth Hypothesen der Beindaten nach der in Abschnitt 3.1 beschriebenen

---

Methode gelabelt und unterteilt (siehe Tabelle 5.1). Der Datensatz **PS** (**P**ositiv **S**eparierbar), beinhaltet dabei die Ground Truth Hypothesen, bei denen sich die Beine eindeutig trennen lassen. Der Datensatz **PV** (**P**ositiv **V**erdeckt) die, bei denen sich die Beine gegenseitig verdecken. Der Datensatz **PM** (**P**ositiv **M**erged) enthält Daten, bei denen die Beine durch die Segmentierung zu einem Segment zusammengefasst wurden. Der Datensatz **PBein**, ist die Vereinigung dieser drei Teildatensätze. Da sich die Häufigkeiten der Erscheinungsformen aufgrund der unterschiedlichen Sensorhöhen stark unterscheiden, wurden die Nachfolgenden Tests nur mit den Daten des Lasersensors auf 40cm Höhe durchgeführt. Die Ergebnisse für den Sensor auf 23cm Höhe können in Anhang A.3 gefunden werden.

Die Datensätze für Rollstühle (**PRollstuhl**), Rollatoren (**PRollator**) und Krücken (**PKrücken**) wurden ebenfalls mit dem Lasersensor auf 40cm Höhe aufgenommen. Da diese ebenfalls mit den Mitarbeitern des Fachgebietes erzeugt wurden, beinhalten diese keine natürliche Benutzung dieser Hilfsmittel.

	<b>PBein</b>	%	<b>PS</b>	%	<b>PV</b>	%	<b>PM</b>	%
23cm Sensorhöhe	15230	100%	8355	55%	2583	17%	4292	28%
40cm Sensorhöhe	12976	100%	4377	34%	3186	25%	5413	41%

**Tabelle 5.1:** Auflistung der Ground Truth Hypothesen, unterteilt nach den definierten Klassen. Es ist zu erkennen, dass sich mit zunehmender Sensorhöhe Beine weitaus schlechter separieren lassen.

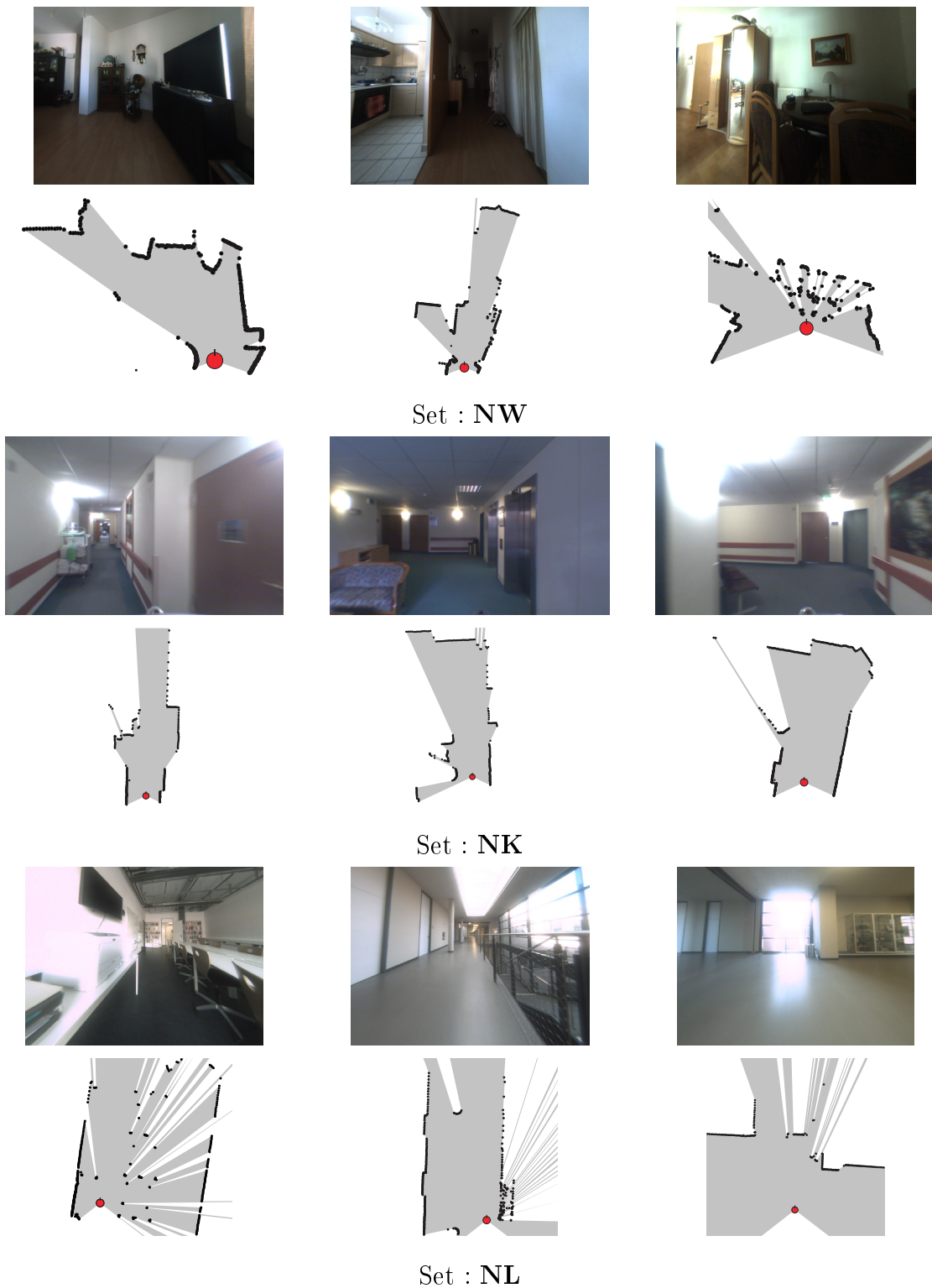
## Negativdaten

Die Negativdatensätze wurden ebenso in drei verschiedenen Gruppen unterteilt, je nachdem in welcher Umgebung sie aufgenommen wurden.

Das erste Negativset **NW** (**N**egativ **W**ohnraum) wurde in privaten Seniorenwohnungen der AWO<sup>1</sup> in Erfurt aufgenommen. Bei diesem Datensatz ist zu erkennen, wie gut die Verfahren für natürlichen menschlichen Lebensraum geeignet sind. Er besteht aus den Wohnungen vier älterer Menschen sowie eines Gemeinschaftsraumes und enthält

---

<sup>1</sup>Arbeiterwohlfahrt



**Abbildung 5.1:** Beispiele für die Negativdatensätze

*NW Wohnraum, NK Krankenhaus, NL Labor. Kameraaufnahme und korrespondierende Laserscans*

viele Stühle, Tische, Sessel und sonstige Möbelstücke auf begrenzten Raum.

Das zweite Negativset **NK** (**N**egativ **K**rankenhaus) wurde in der m&i-Fachklinik Bad Liebenstein hauptsächlich im ersten und zweiten Obergeschoss aufgenommen und soll typische Krankenhaussituationen repräsentieren. Enthalten sind hauptsächlich lange Gänge mit vereinzelt Sitzmöglichkeiten und Wagen für Reinigungsarbeiten, die am Aufnahmetag durchgeführt wurden. Es wurde darauf geachtet, dass keine Rollstühle oder Rollatoren in diesem Datensatz enthalten sind.

Der Dritte und wie sich zeigte schwierigste Datensatz **NL** (**N**egativ **L**abor), wurde im Zusebau der TU-Ilmenau aufgenommen. Er enthält Konferenzräume mit vielen Stühlen, weitläufige Freiflächen mit Säulen, Gitter an Treppenaufgängen und große Glasscheiben, welche die Laserstrahlen in andere Richtungen reflektieren können. Beispielbilder und Laserscans zu den Datensätzen können in Abbildung 5.1 gefunden werden können.

### 5.3 Evaluierungsmethodik

Für die Evaluierungen der Testergebnisse wurden zwei Methoden gewählt. Um die GDIF Merkmale direkt mit den Merkmalen von Arras zu vergleichen, wurde für die einzelnen Segmente die Entscheidung des Klassifikators mit den Ground Truths der Datensätze verglichen. Hierzu wurden aus der Ausgabe des Adaboost Strong Learners und verschiedener Thresholds ROC Kurven (**R**eciever **O**perating **C**haracteristic) berechnet. Diese setzen die True Positive Rate (TPR) mit der False Positive Rate (FPR) ins Verhältnis. Um diese Kurven mit einem Wert besser vergleichbar zu machen, wurde die Ballanced Error Rate (BER) über alle Thresholds berechnet und der Beste ermittelte Wert als Vergleichsmaß verwendet. Dieser entspricht dem nord-westlichsten Punkt der Kurve und ist somit der Threshold, ab dem bei einer weiteren Erhöhung mehr falsche Klassifikationen als richtige hinzukommen würden. Für einen Beispielgraph siehe Abbildung 5.2, für die Formeln zur Berechnung siehe Tabelle 5.2.

Die endgültigen Personenhypothesen hängen allerdings nicht alleine von der Klas-

---

sifikation ab. So können die Anzahl der erzeugten Segmente und die verwendete Heuristik für die Personenhypothese die Ergebnisse stark beeinflussen. Um möglichst aussagekräftige Ergebnisse zu erstellen, wurde für diese Evaluierungen ein Precision-Recall Graph berechnet. Dieser setzt die Rate aller erkannten Ground Truth Personenhypothesen (Precision) mit der Rate aller korrekt erkannten Personenhypothesen (Recall) ins Verhältnis. Um diese Ergebnisse einfacher vergleichbar zu machen, wurde der F1 Score über alle Thresholds berechnet und wiederum der Beste als Vergleichsmaß verwendet. Dieser ist ebenso wie der BER für die ROC-Kurven der Punkt in der Kurve, ab dem bei einer weiteren Erhöhung der Thresholds mehr Falsche Personenhypothesen als korrekte hinzukommen würden. Ein Beispielgraph kann in Abbildung 5.2 gefunden werden. Für die Formeln zur Berechnung siehe Tabelle 5.2.

$TPR$	$= \frac{TP}{TP + FN}$	$Precision$	$= \frac{TP}{TP + FP}$
$FPR$	$= \frac{FP}{FP + TN}$	$Recall$	$= \frac{TP}{TP + FN}$
$BER$	$= \frac{1}{2} \left( \frac{FN}{P} + \frac{FP}{N} \right)$	$F1\ Score$	$= \frac{2TP}{2TP \cdot FP \cdot FN}$

**Tabelle 5.2:** Formeln für die Evaluierung.

**Links :** Für die Klassifikation von Segmenten. **Rechts :** Für die Personenhypothesen.

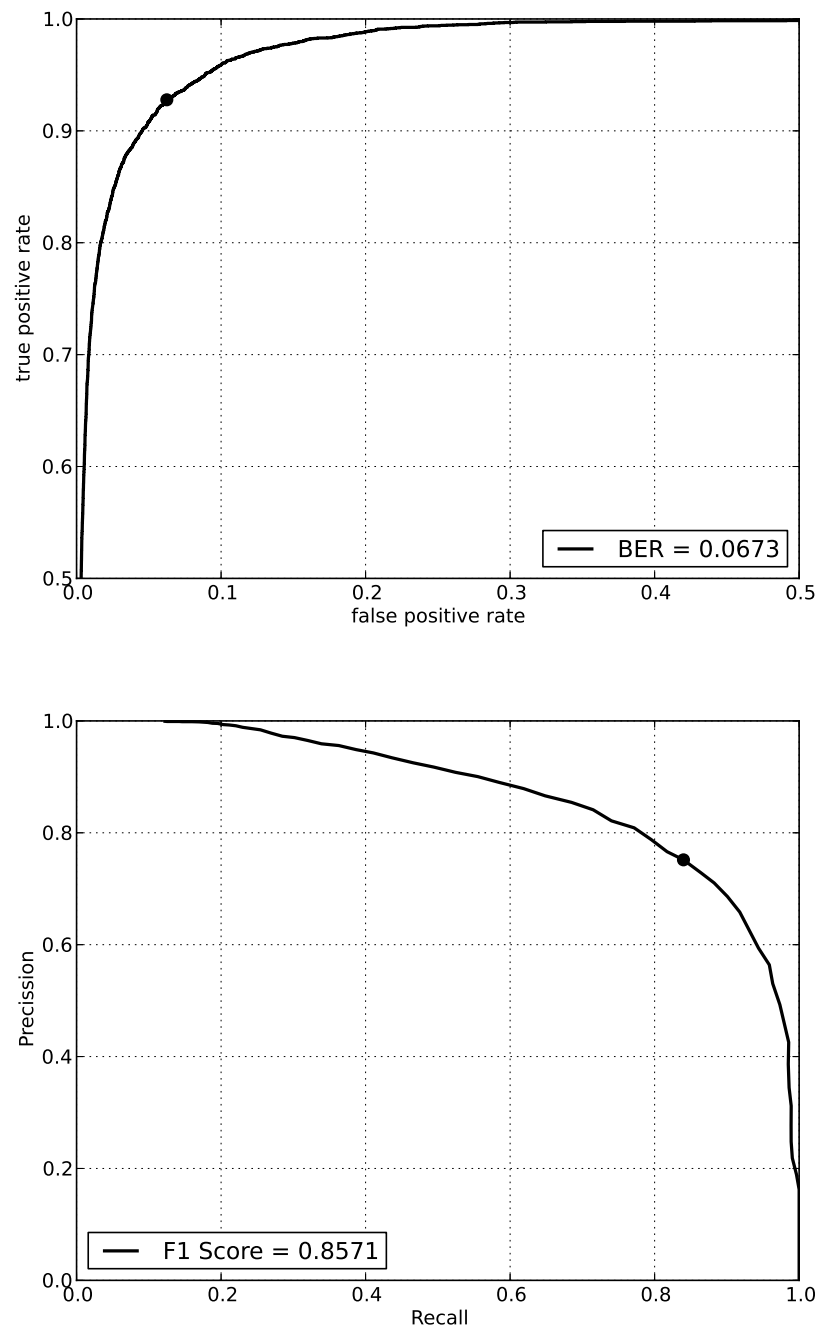
### Einschränkungen

Aufgrund der spezifischen Umstände, die sich aus der Natur von Laserdaten ergeben, wurden einige Einschränkungen getroffen um die Verfahren möglichst objektiv zu testen. Da der Lasersensor in relativ schneller Abfolge Daten produziert, können sich

aufeinander folgende Scans stark ähnlich sein. Deshalb wurden negative Trainingsdaten nur dann in den Datensatz aufgenommen, falls sich der Roboter um einen bestimmten Abstand zur vorherigen Messung bewegt hatte. Ebenso wurden, um eine mögliche Überanpassung an einen bestimmten Datensatz nicht in die Evaluierung einfließen zu lassen, Trainings- und Testdaten streng zeitlich getrennt. Des weiteren, existiert in den positiven Datensätzen Verfahrensbedingt ein statischer Hintergrund. Experimente zeigten, dass sich dieser sehr gut erlernen lässt. So erreichte ein auf einen Datensatz mit statischen Hintergrund trainierter Klassifikator nahezu perfekte Ergebnisse auf Trainings- und Testdaten. Um dadurch die Evaluation nicht zu verfälschen, wurde der Hintergrund der positiven Datensätze weder für das Training verwendet, noch wurden falsch positive Detektionen für diese Datensätze gezählt. Damit eine Detektion als korrekt gewertet wird, muss sich diese innerhalb der halben maximal möglichen Beinabstandes von 35cm zu der Ground Truth Position befinden. Alle Datensätze wurden in einem Verhältnis von ca. 80% zu 20% in Trainings- und Testdaten aufgeteilt. Wobei die Positivdatensätze jeweils nur eine Person pro Laserscan enthalten. Wobei zum Ende dieser Masterarbeit leider kein geeignet großer Datensatz für Personen mit Krücken zur Verfügung stand, um sinnvolle Tests durchzuführen

	# GT	%	# Train	%	# Test	%
<b>PBein</b>	12976	100%	10181	78%	2795	22%
<b>PRollator</b>	3496	100%	2798	80%	698	20%
<b>PRollstuhl</b>	3684	100%	2972	81%	712	19%

	# Scans	%	# Train	%	# Test	%
Set <b>NW</b> (Negativ Wohnraum)	6227	100%	4846	78%	1381	22%
Set <b>NK</b> (Negativ Krankenhaus)	7767	100%	6184	80%	1583	20%
Set <b>NL</b> (Negativ Labor)	17157	100%	13491	79%	3666	21%
Set <b>NALL</b> (Gesamt)	31151	100%	24521	79%	6630	21%



**Abbildung 5.2:** Evaluierungsgraphen für die Klassifikation

***Oben :** für Segmenten. **Unten :** für Personenhypothesen. Bester BER und F1 Score durch Punkte auf den jeweiligen Graphen gekennzeichnet.*



## 5.4 Vergleich zu Arras et. al.

Dieser Abschnitt soll zunächst einen Überblick geben, wie gut die neu vorgestellte Merkmale im Vergleich zum State of the Art Ansatz sind. Hierzu werden die GDIF Merkmale mit einer Vergleichsimplementierung des Arras Verfahrens verglichen, sowie mit den Ergebnissen die er in seiner Veröffentlichung vorstellte.

Für die Evaluierung seines Verfahrens, verglich Arras die Erkennungsraten der Segmente, welche durch Personen erzeugt wurden und welche nicht durch Personen erzeugt wurden. Dazu teilte er seinen Datensatz ebenso in Trainings- und Testdaten. Die ersten zwei Tests erfolgten, indem er seinen Klassifikator auf einer Korridorumgebung und einer Büroumgebung trainierte und auf den selben wieder testete. Hierbei erhielt er sehr gute Ergebnisse, welche mit der verwendeten Vergleichsimplementierung nicht reproduziert werden konnten. Dies kann mehreren Ursachen haben. Zum einen können die Daten durch eine andere Sensorhöhe und den verwendeten Umgebungen schwerer zu Klassifizieren sein. Zum anderen könne durch die Evaluierungsmethodik des streng zeitlich getrennten Testdatensatzes vereinzelt Objekte in diesem enthalten sein, welche nicht trainiert wurden. Um die Möglichkeit einer unter Umständen fehlerhaften Implementierung auszuschließen, wurde diese mit der Open Source Implementierung von Spinello [SPINELLO und SIEGWART, 2008] verglichen. Hierbei wurden auf den ebenfalls bereitgestellten Daten<sup>2</sup> nahezu exakt die selben Ergebnisse erzeugt, welche auch von dem ebenfalls bereitgestellten Testtool erzeugt wurden. Deshalb kann von einer korrekten Implementierung ausgegangen werden.

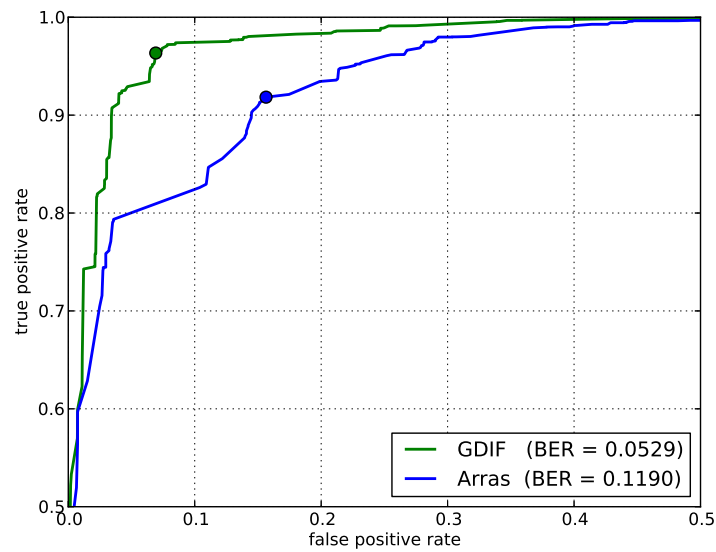
### Ergebnisse mit gleichen Testparametern

Die Ergebnisse für ein Training mit den selben Adaboost-parametern, wie sie Arras verwendete, können in Abildung 5.3 betrachtet werden. Die Baumtiefe der Weak Learner wurde dafür auf eins gesetzt, womit sie denen von Arras entsprechen. Er verwendete für sein Training genau 10 Weak Learner, was er als das Optimum identifizierte. In einem weiteren Test auf den hier verwendeten Daten, konnten aber die

---

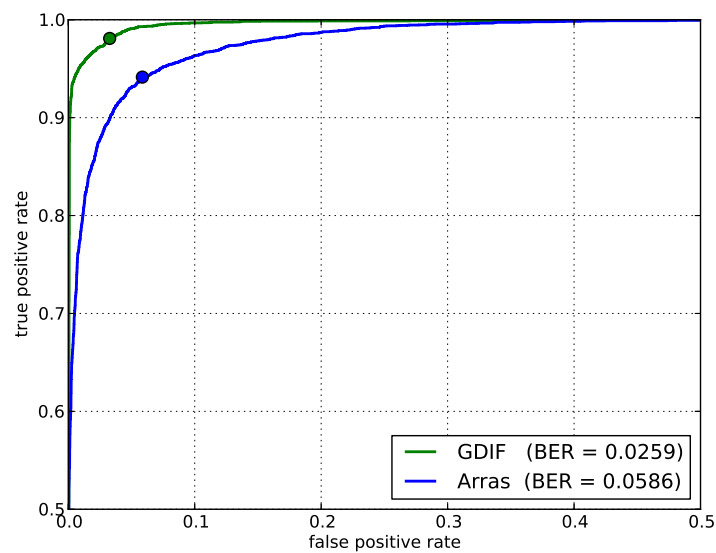
<sup>2</sup><http://www.informatik.uni-freiburg.de/spinello/people2D.html>

---



**Abbildung 5.3:** Testergebnisse für die Segmentklassifikation mit 10 weak learnern und einer Baumtiefe von 1.

*Diese Parametereinstellungen entsprechen dem Klassifikator, wie ihn Arras verwendete. Evaluiert auf Krankenhausdaten.*



**Abbildung 5.4:** Testergebnisse für die Segmentklassifikation mit 50 weak learnern und einer Baumtiefe von 10.

*Mit diesen Parametereinstellungen wurden bei den Tests die besten Ergebnisse erzielt. Evaluiert mit Krankenhausdaten.*

Klassifikatoren durch die Verwendung von 50 Weak Learnern leicht verbessert werden. Auf eine BER von 0,047 für die GDIF Merkmale und 0,090 für die Merkmale von Arras.

### Verbesserung durch decision trees

Durch die Verwendung von Decision Trees, kann die Klassifikationsgüte beider Varianten weiter gesteigert werden. In Abbildung 5.4 sind die Ergebnisse für ein Training mit einer maximalen Baumtiefe von 10 und 50 Weak Learnern zu sehen. Nimmt man auch hier den BER als Maß, konnte die Klassifikationsgüte durch die Verwendung von Decision Trees verdoppelt werden. Diese Trainingsparameter werden auch für die Evaluation der Personenhypothesen verwendet.

### Ergebnisse aller Testumgebungen

	GDIF	Arras
Krankenhaus	0,0259	0,0586
Labor	0,0569	0,0909
Wohnraum	0,0184	0,0661
Gesamt	0,0472	0,0892

**Abbildung 5.5:** Vergleich des BER auf allen Testumgebungen.

*Es ist zu sehen, dass die GDIF Merkmale immer bessere Ergebnisse erzielen.*

### Test der Generalisierungsfähigkeit

Ein dritte Test den Arras durchführte, sollte die Generalisierungsfähigkeit seines Verfahrens demonstrieren. Hierzu trainierte er seinen Klassifikator auf der Büroumgebung und teste auf Daten des Korridors. Dabei erreichte sein Klassifikator einen BER von 0,061. Ein gleicher Test wurde mit den neuen GDIF Merkmalen durchgeführt, indem auf der Laborumgebung trainiert und gegen die Krankenhausumgebung getestet wurde. hierbei wurde ebenso ein deutlich besserer BER von 0,035 erreicht (siehe Abbildung 5.3).

	Detected Label		
Ground Truth	Person	No Person	Total
Person	<b>217</b> (90,42%)	<b>23</b> (9,58%)	<b>240</b>
No Person	<b>112</b> (4,28%)	<b>2504</b> (95,72%)	<b>2616</b>

(a)

	Detected Label		
Ground Truth	Person	No Person	Total
Person	<b>4804</b> (96,54%)	<b>173</b> (3,46%)	<b>4977</b>
No Person	<b>681</b> (3,51%)	<b>18343</b> (96,49%)	<b>19024</b>

(b)

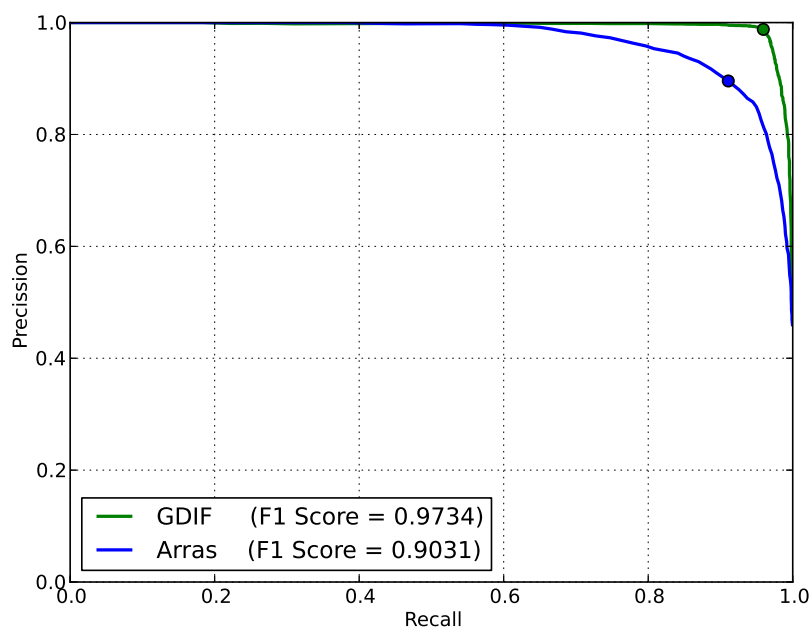
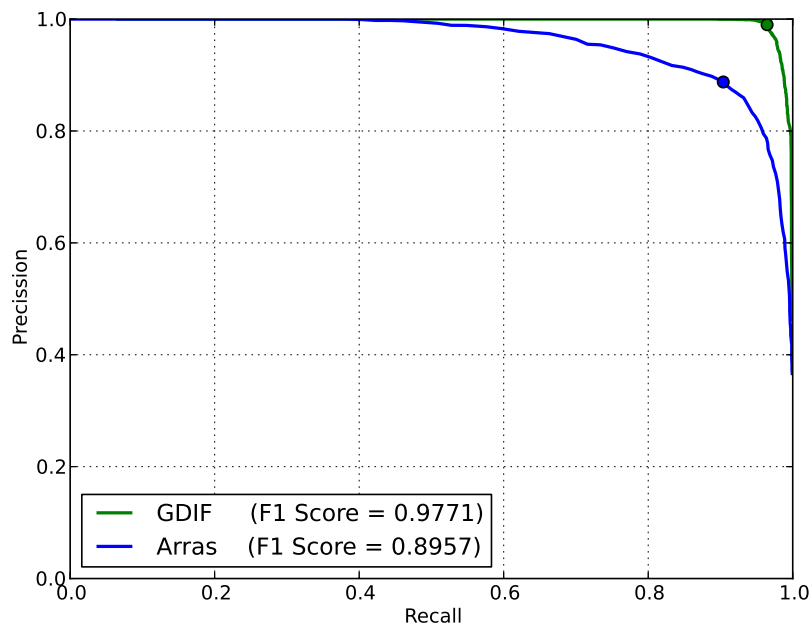
**Tabelle 5.3:** Konfusionsmatrizen von Klassifikatoren mit unterschiedlicher Trainings- und Testumgebung.

(a) Arras Klassifikator (Tabelle V aus [ARRAS et al., 2007]). (b) Klassifikator mit GDIF Merkmalen trainiert mit Labordaten, getestet mit Krankenhausdaten.

## 5.5 Personenhypothesen für Beine

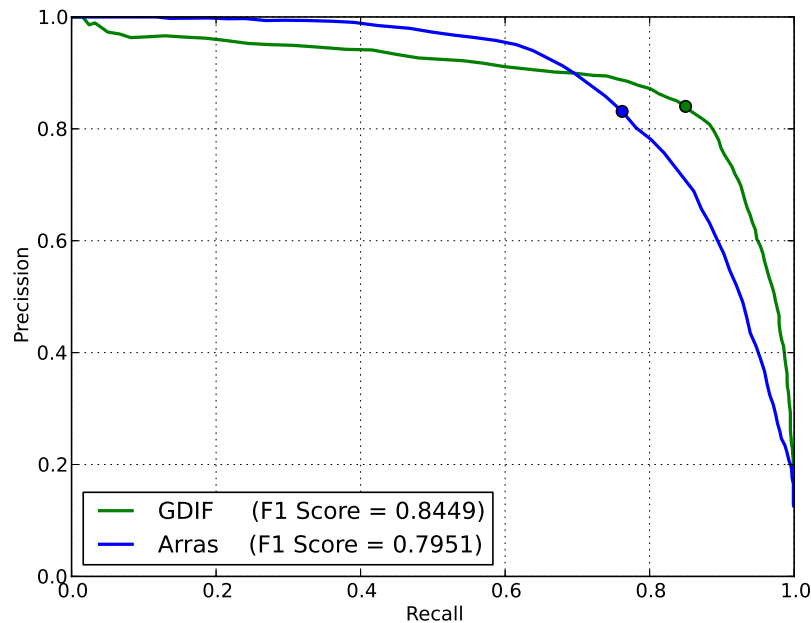
### Ergebnisse für verschiedene Testumgebungen

Bei der Evaluierung auf verschiedenen Negativdatensätzen, zeigten sich teilweise deutliche Unterschiede hinsichtlich der Detektionsgüte. So erzielte die beide Verfahren auf den Wohnraum und Krankenhaus Datensätzen klar bessere Ergebnisse, als auf den Labordaten (siehe Abbildungen 5.9 und 5.7). Eine mögliche Erklärung hierfür ist, dass die Eigenschaft des lokalen Minimums einen sehr starkes Merkmal für Personen ist. So zeigten sich falsch positive Klassifikationen der Laborumgebung vor allem bei offenen Türen, Mülleimern und anderen frei im Raum stehenden Objekten. Auch Glasscheiben und Gitter erzeugten durch teilweise Reflektionen und Abschirmungen solche Situationen. All dies war in den beiden anderen Testumgebungen weitaus weniger vorhanden. Außerdem sind in den Positivdatensätzen meist Personen enthal-



**Abbildung 5.6:** Testergebnisse für die Personenhypothese.

*Oben* : trainiert und getestet mit negativen Wohnraumdaten und PBeine. *Unten* : trainiert und getestet mit negativen Krankenhausdaten und PBeine.



**Abbildung 5.7:** Evaluierungsergebnisse auf dem Krankenhausdatensatz

ten, die nicht von Umgebungsobjekten verdeckt werden und sich in einem gewissen Abstand zum Hintergrund befinden. Deshalb scheint dies als mögliche Erklärung plausibel.

### Ergebnisse für die unterschiedlichen Erscheinungsformen von Personen

Um herauszufinden mit welchen Erscheinungsformen von Beinen ein Klassifikator trainiert werden sollte, um die besten Ergebnisse zu erzielen, wurden Tests mit den Subsets des positiven Trainingsdatensatzes durchgeführt. Dafür wurde der Klassifikator jeweils mit den Daten der Subsets PS, PV, PM und PBein trainiert und mit den Ground Truth Hypothesen des gesamten Testdatensatzes verglichen. In Tabelle 5.4 sind die Ergebnisse dieser Tests zu finden. In den einzelnen Zellen dieser Tabelle, sind die Raten der richtig erkannten Personenhypothesen eingetragen, für einen Threshold, der den besten F1 score über den gesamten Datensatz erzeugt hat. Es ist zu erkennen, dass sowohl bei den Merkmalen von Arras, als auch bei den GDIF Merkmalen am besten mit dem

Train \ Test		GDIF				Arras			
		<i>PS</i>	<i>PV</i>	<i>PM</i>	<i>PBein</i>	<i>PS</i>	<i>PV</i>	<i>PM</i>	<i>PBein</i>
NW	<i>PS</i>	0.9823	0.6156	0.2385	0.7047	0.9114	0.6035	0.3464	0.6856
	<i>PV</i>	0.6832	0.9759	0.6675	0.7513	0.7921	0.9957	0.5158	0.7694
	<i>PM</i>	0.9352	0.9774	0.9749	0.9511	0.8996	0.8418	0.8403	0.8697
	<i>PBein</i>	0.9843	0.9689	0.8918	0.9561	0.9701	0.9279	0.8364	0.9244
NK	<i>PS</i>	0.9836	0.6200	0.2783	0.7077	0.9124	0.6045	0.3364	0.6844
	<i>PV</i>	0.9544	1.0000	0.7730	0.9177	0.7964	0.9915	0.5105	0.7690
	<i>PM</i>	0.9380	0.9632	0.9432	0.9456	0.9024	0.8531	0.7928	0.8613
	<i>PBein</i>	0.9921	0.9844	0.8918	0.9637	0.9644	0.9251	0.7836	0.9070
NL	<i>PS</i>	0.9807	0.6553	0.2862	0.7171	0.9160	0.6836	0.3720	0.7150
	<i>PV</i>	0.6355	0.9929	0.4815	0.6830	0.4448	0.9548	0.4195	0.5639
	<i>PM</i>	0.6526	0.8573	0.8482	0.7547	0.7637	0.7259	0.7124	0.7408
	<i>PBein</i>	0.9231	0.9096	0.6609	0.8505	0.8782	0.8093	0.6569	0.8028
NA	<i>PS</i>	0.9836	0.6200	0.2783	0.7077	0.9124	0.6045	0.3364	0.6844
	<i>PV</i>	0.5793	0.9858	0.4722	0.6513	0.4113	0.9435	0.3878	0.5363
	<i>PM</i>	0.6007	0.8220	0.8218	0.7136	0.7729	0.6991	0.6794	0.7300
	<i>PBein</i>	0.9088	0.9096	0.5976	0.8268	0.8797	0.7711	0.6094	0.7816

**Tabelle 5.4:** Überblick zu allen durchgeführten Tests für die Erscheinungsformen von Beinen.

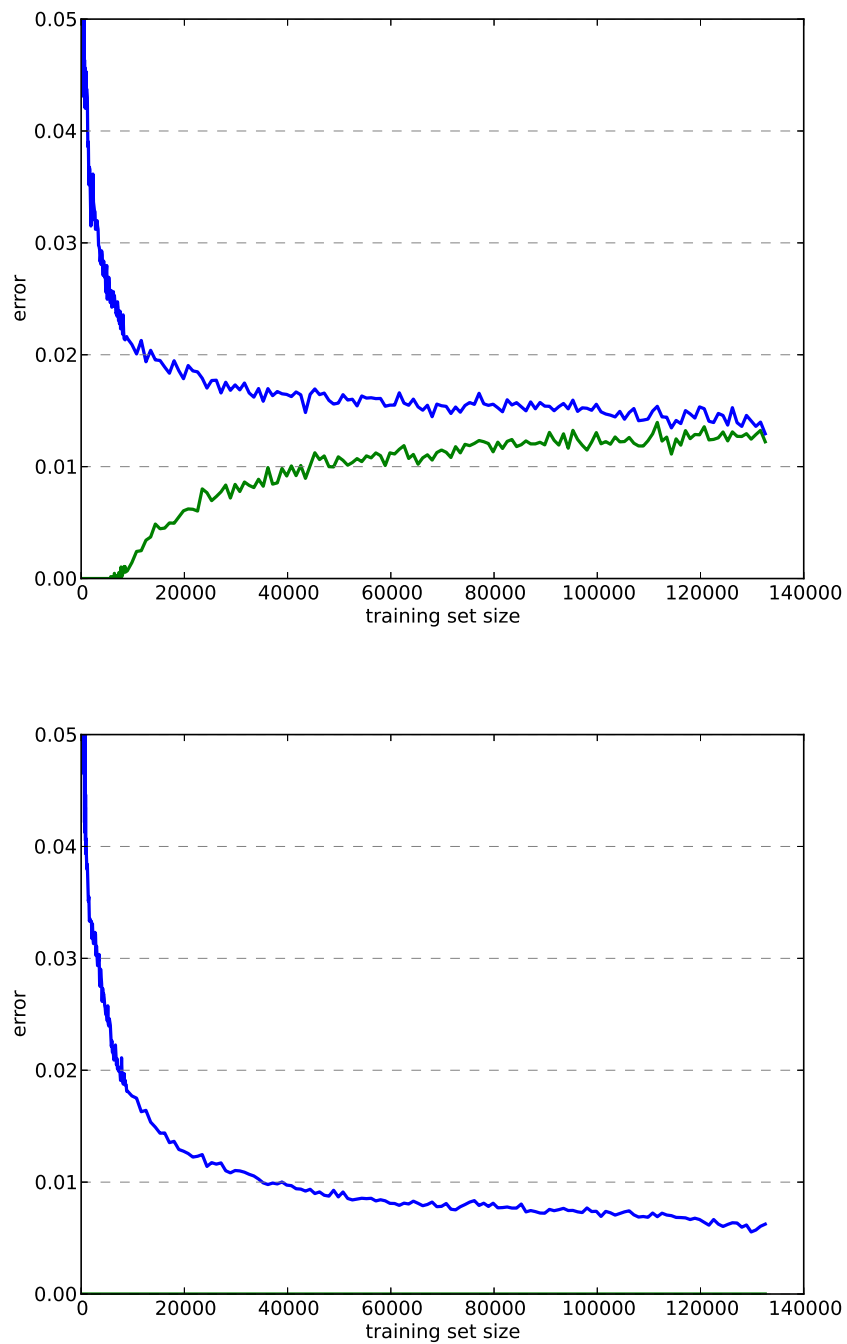
*Trainingsdatensätze in den Zeilen, Validierungsdatensätze in den Spalten. Die in den Zellen enthaltenen Werte entsprechen dem Recall an dem Punkt der Kurve mit dem besten ermittelten F1 score der Personenhypothesen.*

kompletten Positivdatensatz trainiert werden sollte. Obwohl die einzelnen Subsets unterschiedlich schwer zu klassifizieren sind, wurden die besten Ergebnisse erzielt, wenn mit allen Erscheinungsformen trainiert wurde.

## 5.6 Learning Curve Analyse

Bei der Analyse der Learning Curves zeigte sich, dass der Default Parameter für die maximale Baumtiefe von 3 einen relativ hohen Bias produzierte (siehe Abbildung 5.8 oben). Deshalb wurde die Komplexität des Klassifikators schrittweise durch die Erhöhung der Baumtiefe gesteigert. Dabei stellte sich bei einer Baumtiefe von 10 ein Optimum ein (siehe Abbildung 5.8 unten). Obwohl dieser Klassifikator eine sehr hohe Varianz aufweist, scheint sich dies nicht auf die Güte der Erkennungsrate auszuwirken. In einem weiteren Test zeigte sich, dass selbst die Hinzunahme weiterer Merkmale das Training verbessert. So konnte durch die gleichzeitige Verwendung der Merkmale von Arras und GDIF der Fehler noch weiter leicht verringert werden. Dies widerspricht der Lehrmeinung und kann nur durch die Eigenschaften von Adaboost und/oder Ensemble Verfahren im allgemeinen erklärt werden. Anhand der Grafiken kann auch gesehen werden, dass für einen ausreichend guten Klassifikator mindestens 40000 Trainingssamples verwendet werden sollten. Wobei hier ca. 4000 positive und 36000 negative Samples im Training enthalten waren. Insgesamt wurden für durchgeführten Tests 20539 positive Beinsample und 158599 negative Samples verwendet. Durch eine weitere Vergrößerung des Datensatzes könnten die Werte aber durchaus noch gesteigert werden.





**Abbildung 5.8:** Learning Curves mit unterschiedlicher maximaler Baumtiefe der weak learner und GDIF Merkmalen.

*Oben : maximale Baumtiefe von 3. Unten : maximale Baumtiefe von 10.*

Verfahren	GDIF 50-1	GDIF 50-10	Arras 50-1	Arras 50-10	Spinello 50-1	Spinello 50-10
Vorverarbeitung	1.1ms	1.1ms	1.1ms	1.1ms	1.1ms	1.1ms
Segmentierung	1.7ms	1.7ms	1.7ms	1.7ms	1.7ms	1.7ms
Merkmale	0.5ms	0.5ms	10.7ms	10.7ms	1.3ms	1.3ms
Klassifikation	0.1ms	0.3ms	0.1ms	0.3ms	0.1ms	0.3ms
Hypothese	10 $\mu$ s	10 $\mu$ s	10 $\mu$ s	10 $\mu$ s	10 $\mu$ s	10 $\mu$ s
Overhead	3.0ms	3.0ms	2.9ms	2.9ms	3.1ms	3.1ms
Gesamt	6.4ms	6.6ms	16.5ms	16.7ms	7.3ms	7.5ms

**Tabelle 5.5:** Laufzeiten unterschiedlicher Merkmale und Klassifikatoren pro Laserscan

*Trainiert und Evaluiert auf Krankenhausdaten und einer 3000MHz CPU*

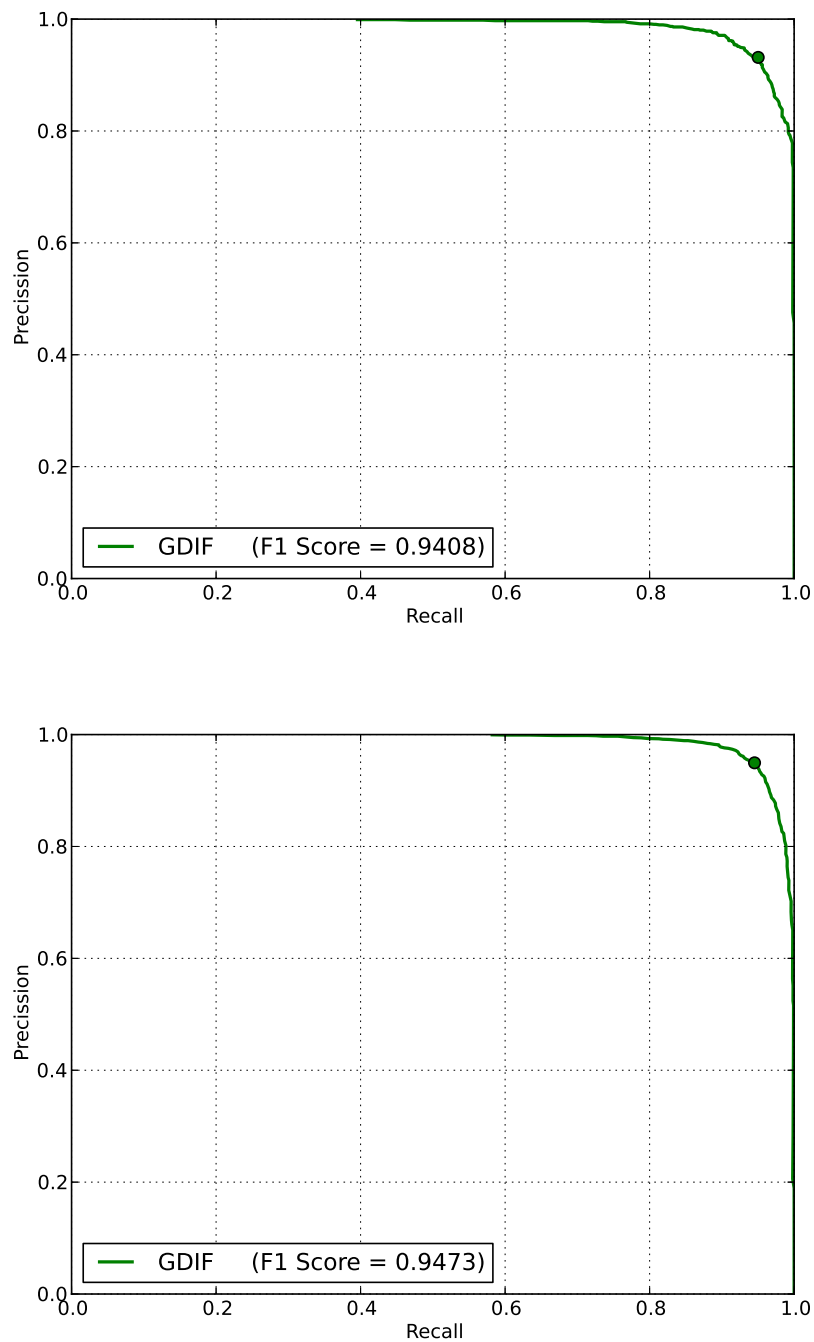
## 5.7 Laufzeit

In Tabelle 5.5 sind die Ergebnisse für die Laufzeitanalyse zu sehen. Hierzu wurden verschiedenen Klassifikatoren mit 50 Weak Learnern und einer Baumtiefe von 1 bzw. 10 trainiert. Dabei wurden unterschiedliche Merkmale verwendet. Zum einen die neu implementierten GDIF Merkmalen, die Merkmale von Arras, welche bereits am Fachgebiet implementiert waren und die Merkmale von Arras, welche von Spinello Open Source zur Verfügung standen. Es wurden die Laufzeiten der einzelnen Schritte aufgelistet. Im Overhead sind Dinge wie das Einlesen der Daten und Funktionsaufrufe enthalten. Dabei ist zu erkennen, dass die am Fachgebiet implementierten Arras Merkmale deutlich langsamer sind, als die Implementierung von Spinello. Dies kann einerseits an einer ineffizienten Implementierung liegen, andererseits daran dass Spinello leicht unterschiedliche Merkmale verwendete. Dennoch sind die neu implementierten GDIF mehr als doppelt so schnell berechenbar wie die Implementierung von Spinello. So kann mit ihnen ein kompletter Laserscan innerhalb von 6,6ms Verarbeitet werden. Da der Laserscanner nur alle 100ms neue Daten liefert, ist die Echtzeitfähigkeit des Verfahrens gewährleistet.

## 5.8 Hypothesen für Rollstühle und Rollatoren

Für die Detektion von Rollstühlen und Rollatoren konnten ebenso sehr gute Ergebnisse erzielt werden. Die Ergebnisse können in 5.9 betrachtet werden. Anzumerken hierbei ist, dass diese immer noch besser erkannt werden konnten, als Personen die durch ihre Beine in Erscheinung treten, wenn die Merkmale von Arras verwendet werden.

---



**Abbildung 5.9:** Evaluierungsergebnisse für Rollstühle und Rollatoren.

*Oben : Rollatoren. Unten : Rollstühle. Evaluiert mit negativen Krankenhausdaten.*



## Kapitel 6

# Zusammenfassung und Ausblick

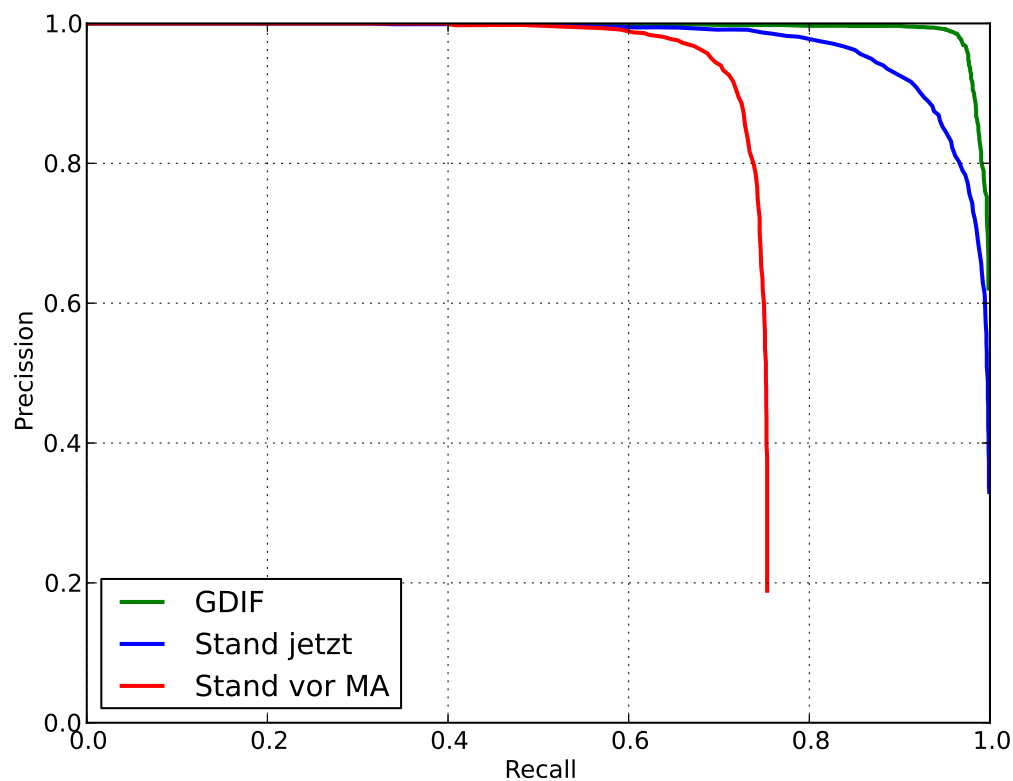
### 6.1 Zusammenfassung

In dieser Masterarbeit wurde ein neues Verfahren für die Merkmalsgenerierung zur Detektion von Personen in Laserscans vorgestellt. Mit diesen Merkmalen konnten generell (und teilweise deutlich) bessere Ergebnisse erzielt werden, als mit dem State of the Art Ansatz. Zusammen mit der Erstellung eines geeigneten Trainingsdatensatzes, der Analyse der Ergebnisse und der Implementierung einer alternativen Heuristik zur Generierung der Personenhypothese, konnte die Detektionsgüte des am Fachgebiet eingesetzten Personendetektors erheblich gesteigert werden (siehe Abbildung 6.1)).

Ebenso ist es mit den neuen GDIF Merkmalen auch möglich abweichende Erscheinungsformen von Personen zu erkennen. So konnte gezeigt werden, dass Personen mit Rollatoren und in Rollstühlen so gut erkannt werden können, sodass dieses Verfahren in der Praxis anwendbar ist.

Wichtige Erkenntnisse dieser Masterarbeit sind unter anderem, dass Personenhypothesen schon gebildet werden sollten, wenn bereits ein Beinsegment als positiv klassifiziert wurde. Dabei sollten für das Training alle Erscheinungsformen von Personen verwendet werden. Für einen genügend guten Klassifikator sollten mindestens 40000 Trainings-samples aufgenommen werden. Wobei mindestens 10% positive Samples sein sollten. Durch eine weitere Vergrößerung des Datensatzes, kann die Klassifikationsgüte aber

---



**Abbildung 6.1:** Vergleich vom Stand vor der Masterarbeit und jetzt.

*Evaluert auf Krankenhausdaten. Anmerkung : die Precission kann nicht 0 werden falls der Recall größer 0 ist.*

durchaus gesteigert werden.

## 6.2 Ausblick

### weiterführende Tests

Für das Training wurden bisher nur positive Samples verwendet, die nur Messpunkte innerhalb der Box enthielten welche durch Personen erzeugt wurden. Bei der Evaluierung zeigte sich, dass das neu implementierte Verfahren auf dem Negativdatensatz mit Wohnraumumgebung sehr gute Ergebnisse erzielte. Dieser besteht hauptsächlich aus Lasermessungen mit enger räumlicher Begrenzung. Da in solchen Umgebungen aber

weit häufiger Situationen vorkommen können, bei denen Personen nah an Umgebungsobjekten stehen, werden für eine objektive Einschätzung in solchen Umgebungen weitere Tests benötigt. Wofür aber für diese Masterarbeit keine geeigneten Daten zur Verfügung standen.

Außerdem liegen keine Erkenntnisse darüber vor, wie sich gut das Verfahren mit Szenen zurecht kommt, in denen mehrere Personen enthalten sind. Da in den verwendeten Trainingsdaten jeweils nur eine Person enthalten war.

### **alternative Hypothesengenerierung**

Für die Personenerkennung mittels Beinpaardetektion, wurde ein relativ einfacher Ansatz gewählt. Dieser nimmt willkürlich ein positiv klassifiziertes Segment, und sucht nach einem weiteren innerhalb des maximal möglichen Beinabstands. Hier könnten weitaus fortschrittlichere Verfahren, wie die Ungarische Methode zum Einsatz kommen.

### **Analyse weiterer Merkmale**

Es zeigte sich, dass wenn die Merkmale die in dieser Masterarbeit vorgestellt wurden mit den vorhandenen kombiniert werden die Detektionsgüte einen weiteren Anstieg erfährt. in weiterführenden Arbeiten könnte dies weiter analysiert werden. Mittels Kaskadierung des Adaboost Klassifikators, könnte dabei der Geschwindigkeitsvorteil der neu vorgestellten Merkmale beibehalten und der Zuwachs der Detektionsgüte in einem weiteren Kaskadierungsschritt erfolgen. Desweiteren, könnte es sich auch lohnen die Merkmale zu testen, die in anderen Arbeiten als Erweiterungen des Ansatzes von Arras vorgestellt wurden [SPINELLO et al., 2010, XAVIER et al., 2005, PREMEBIDA et al., 2009].

### **Anwendung auf 3D Daten**

Eine interessanter Ansatz wäre, die GDIF Merkmale in Zusammenhang mit 3D Laserscannern oder 3D Kameras wie der Kinect zu testen. Dabei könnten sie mit einigen in Abschnitt 2.2.2 vorgestellten Verfahren kombiniert werden. Eine Verbesserung der Ergebnisse wäre durchaus anzunehmen.

---





# Anhang A

## weitere Ergebnisse

### A.1 Vergleich der Adaboost Varianten

Bei Tests mit unterschiedlichen Adaboost Varianten zeigte sich, dass Discrete-, Logit- und Gentle-Adaboost nahezu gleichgute Ergebnisse lieferten (siehe Abbildung A.1). Wobei die Discrete und Gentle Variante immer ein Bisschen besser waren als Logit. Real-Adaboost hingegen war immer und teilweise deutlich schlechter. Weshalb auf diese Variante für die Personendetektion nicht verwendet werden sollte.

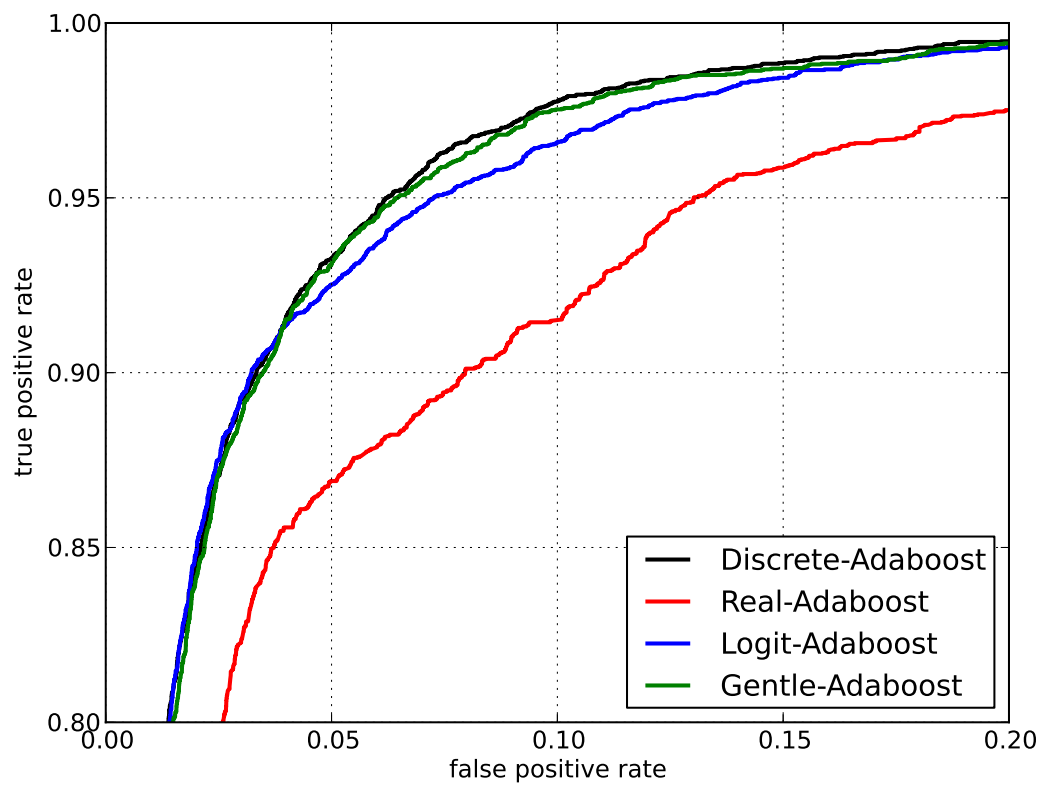
### A.2 Vergleich der Hypothesengenerierung

Wie in 3.2.5 beschrieben, können mehrere Möglichkeiten verwendet werden um eine Personenhypothese aus klassifizierten Beinsegmenten zu generieren. Hierbei zeigte sich, dass der neu implementierte Algorithmus (blaue Linie in Abbildung A.2) dem zuvor verwendeten (rote Linie) deutlich überlegen ist.

### A.3 Ergebnisse für den Roboter SCITOS G3

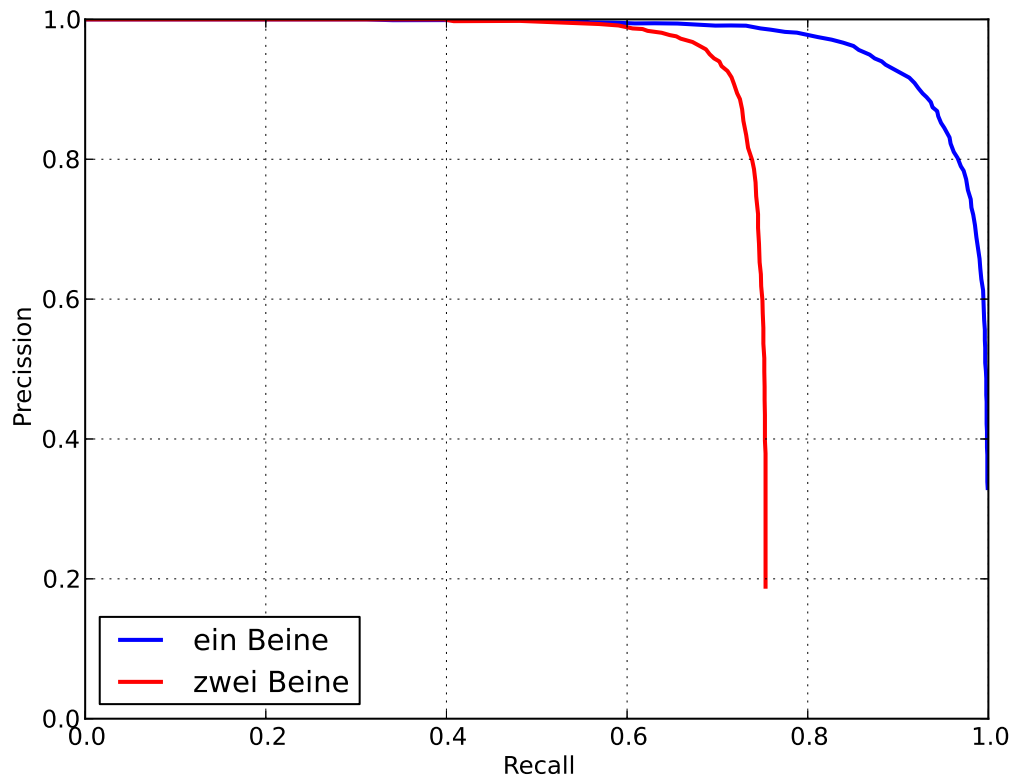
Die gleichen Experimente für die Personendetektion durch Beine wie in Abschnitt 5.5, wurden ebenso mit den Daten des niedriger gelegenen Lasersensors des Roboters SCITOS G3 durchgeführt. Siehe hierzu Tabelle A.1.

---



**Abbildung A.1:** Vergleich unterschiedlicher Adaboost Varianten für die Beinde-  
tektion.

*Evaluert auf Labordaten mit GDIF Merkmalen.*



**Abbildung A.2:** Vergleich der Beiden Methoden zur generierung der Personenhypothese.

**Rot :** es werden zwei Beine benötigt um eine Personenhypothese zu generieren **Blau** : ein Bein reicht aus für eine Personenhypothese. Klassifikator mit Arras Merkmalen. Anmerkung : die Precision kann nicht 0 werden falls der Recall größer 0 ist.

Train \ Test		GDIF				Arras			
		$PS_V$	$PV_V$	$PM_V$	$PA_V$	$PS_V$	$PV_V$	$PM_V$	$PA_V$
NW	$PS_T$	1.0000	0.9933	0.9131	0.9712	0.9832	0.9069	0.6659	0.8684
	$PV_T$	0.8178	0.9933	0.9567	0.8781	0.7831	0.9875	0.4068	0.7096
	$PM_T$	0.9920	0.9889	0.9786	0.9730	0.9256	0.8736	0.6358	0.8413
	$PA_T$	0.9988	0.9867	0.9479	0.9836	0.9641	0.9257	0.6500	0.8752
NK	$PS_T$	0.9998	0.9911	0.9021	0.9579	0.9554	0.9246	0.6341	0.8455
	$PV_T$	0.9544	1.0000	0.7731	0.9178	0.7964	0.9915	0.5106	0.7691
	$PM_T$	0.9803	0.9894	0.9844	0.9536	0.9405	0.9170	0.6965	0.8222
	$PA_T$	0.9900	0.9734	0.9232	0.9701	0.9722	0.9069	0.6144	0.8440
NL	$PS_T$	0.9512	0.8503	0.4569	0.8059	0.8883	0.7428	0.3368	0.7218
	$PV_T$	0.8505	0.9767	0.5239	0.7799	0.6374	0.9636	0.3387	0.6007
	$PM_T$	0.8954	0.9069	0.7965	0.8705	0.8044	0.7627	0.5085	0.7199
	$PA_T$	0.9151	0.9246	0.7215	0.8979	0.8600	0.8137	0.4750	0.7511
NA	$PS_T$	0.9485	0.8326	0.4377	0.7968	0.8929	0.6907	0.3308	0.7160
	$PV_T$	0.8517	0.9778	0.5162	0.7788	0.7107	0.9501	0.3242	0.6391
	$PM_T$	0.8841	0.9069	0.7926	0.8627	0.7873	0.6984	0.4822	0.6941
	$PA_T$	0.9268	0.9357	0.7500	0.8841	0.8517	0.7738	0.4536	0.7351

**Tabelle A.1:** Ergebnisse für die Daten des Roboters Tweety.

*Trainingsdatensätze in den Zeilen, Validierungsdatensätze in den Spalten. Die in den Zellen enthaltenen Werte entsprechen dem Recall an dem Punkt der Kurve mit dem besten ermittelten F1 score der Personenhypothesen.*

## Anhang B

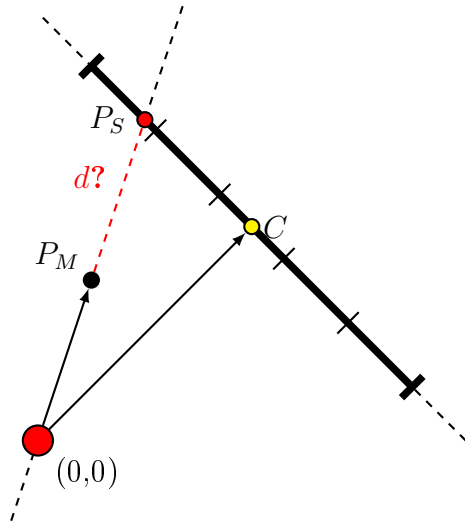
# Berechnungsmethoden für die lokalen Tiefenwerte

Für die Berechnung der Werte, die in die Bins des Tiefenhistogramms eingetragen werden, ist es notwendig den Abstand eines Messpunktes zur Mittellinie der Bounding Box zu bestimmen. Neben der in Kapitel 4 beschriebenen Methode wurden dafür noch weitere Implementiert und getestet.

### **Methode mittels Schnittpunktberechnung**

Eine Möglichkeit den Abstand eines Messpunktes zur Mittellinie der Box zu bestimmen, ist indem man die Mittellinie der Box sowie die Scanlinie des Messpunktes in Geradengleichungen umformt (Abbildung B.1). Mit Hilfe dieser Gleichungen kann ein Schnittpunkt  $P_S$  berechnet werden, der auf beiden Geraden liegt. Dessen Euklidischer Abstand zum Messpunkt kann dann in die Histogramme eingetragen werden kann. Diese Methode ist zwar effizient berechenbar, versagt aber wenn der Anstieg  $m$  einer der beiden Geraden 0 oder  $\infty$  ist. Außerdem können größere Rundungsfehler auftreten, falls der Anstieg gegen die genannten Werte strebt. Eine Möglichkeit diese Nachteile zu umgehen, wäre die Bounding Box und alle relevanten Punkte in einen Teil des Koordinatensystems zu Projizieren bei dem der Anstieg beider Geraden solche Werte nicht annehmen kann. Dieser Ansatz wurde aber aufgrund des entstehenden Overheads an Berechnungsaufwand verworfen.

---



$$g : y = mx + n \quad (\text{B.1})$$

$$g_M : y = m_M x \quad (\text{B.2})$$

$$g_C : y = m_C x + n_C \quad (\text{B.3})$$

$$m_m = \tan(\phi_M) \quad (\text{B.4})$$

$$m_C = \frac{-1}{\tan(\phi_C)} \quad (\text{B.5})$$

$$n_C = y_C - m_C x_C \quad (\text{B.6})$$

$$x_S = \frac{n_C}{m_M - m_C} \quad (\text{B.7})$$

$$d = r_S - r_M \quad (\text{B.8})$$

**Abbildung B.1:** alternative Methode zur Berechnung der lokalen Tiefenwerte

Eine generelle Methode um die Schnittpunkte zweier Geraden zu berechnen, die für jegliche Anstiege funktioniert, ist über ihre impliziten Gleichungen. Die einfachste Möglichkeit diese zu berechnen, ist zwei Punkte auf der Geraden zu suchen (in Homogenen Koordinaten) und das Kreuzprodukt aus ihnen zu bilden.

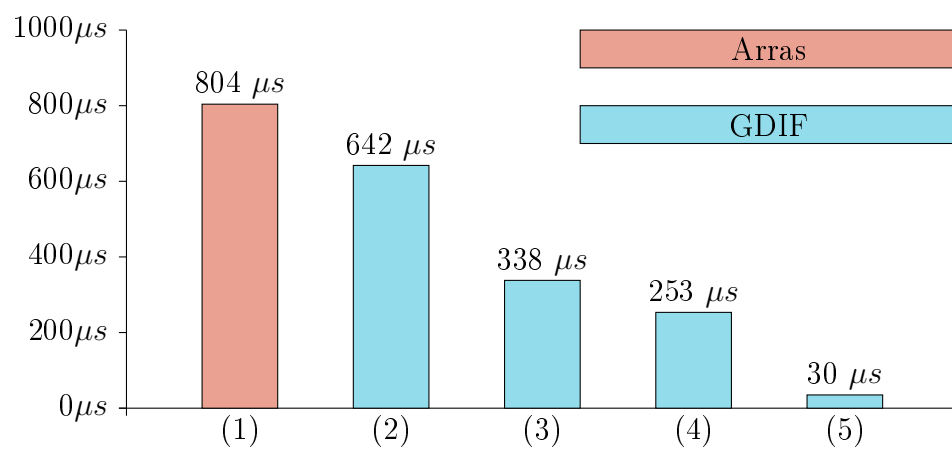
$$L : 0 = ax + by + cw$$

$$L_M := (x_M, y_M, 1) \times (0, 0, 1)$$

$$L_C := (x_C, y_C, 1) \times (0, n_C, 1)$$

$$P_S = L_M \times L_C = \begin{bmatrix} x_S \\ y_S \\ w_S \end{bmatrix}$$

Aus den so entstandenen impliziten Geradengleichungen  $L_M$  und  $L_C$  kann wiederum das Kreuzprodukt gebildet werden um den Schnittpunkt  $P_S$  zu generieren. Dieser ist wiederum in homogenen Koordinaten. Um aus diesen wieder Kartesische zu erzeugen, muss seine  $x$  und  $y$  Komponente durch  $w$  dividiert werden. Diese Methode benötigt zwar keine Sonderbehandlungen und erzeugt somit auch keinen Overhead an Berechnungsaufwand, allerdings ist sie durch die genau 26 Multiplikationsoperationen auch nicht sonderlich effizient. Deshalb wurde auch diese Methode zugunsten der zugunsten der Berechnung mittels trigonometrischer Funktionen und Dreisatz verworfen.



**Abbildung B.2:** Vergleich der durchschnittliche Laufzeiten der Merkmalsberechnung pro Segment

(1) *Bestehende Features*, (2) *Berechnung mittels linesegment Intersection (c++ Boost library)*, (3) *Berechnung mittels expliziter Geradengleichung und Transformation*, (4) *Berechnung mittels impliziter Geradengleichung (unter Verwendung der c++ Eigen library)*, (5) *Berechnung mittels Dreisatz*





## Anhang C

### weitere Algorithmen

---

**Eingaben**


---

```

1    $S = \{s_1, s_2, \dots, s_N\}$  // Segmente des Laserscans
2    $L = \{l_1, l_2, \dots, l_N\}$  // Label des Laserscans
3    $maxLegDist$  // maximal möglicher Abstand zweier Beine

```

**Algorithmus**

```

4   Für jeden  $l_i \in L$ ; // Filter für zu kleine Segmente
5       falls  $|s_i| < 3$ ;
6           setze  $l_i := NOT\_LBELED$ ;
7   Für jeden  $l_i \in L$ ; // setze alle vordegrund segmente als beine
8       falls  $l_i = FOREGROUND$ ;
9           setze  $l_i := SEPARIERT$ ;
10  Für jeden  $l_i \in L$ ; // behandlung verdeckter segmente
11      falls  $l_i = SEPARIERT$  und  $l_{i+1} = SEPARIERT$ ;
12          setze  $l_i := VERDECKT$ ;
13          setze  $l_{i+1} := VERDECKT$ ;
14  Für jeden  $l_i \in L$ ; // behandlung einzelner segmente
15      falls  $l_i = SEPARIERT$ ;
16          falls  $\nexists s_j : dist(s_i, s_j) < maxLegDist, s_j \in S, j \neq i$ ;
17              setze  $l_i := MERGED$ ;

```

**Rückgabe**

```

18   $L' = \{l'_1, l'_d, \dots, l'_N\}$  // modifizierte Label der Segmente

```

---

**Abbildung C.1:** Verwendeter algorithmus zum labeln der Trainingsdaten

---

**Eingaben**


---

```

1    $X = \{(x_1, x_2, \dots, x_m)\}$  // Merkmalsvektoren
2    $L = \{l_1, l_2, \dots, l_m\}$  // Label der Merkmalsvektoren  $l_i \in \{-1, +1\}$ 
3    $m = a + b$  // a Anzahl positiver Samples  $l_i = +1$ 
4   // b Anzahl negativer Samples  $l_i = -1$ 
5    $T$  // Anzahl der Trainingsrunden

```

**Algorithmus**

```

6   Für jedes  $i \in \{1 \dots m\}$ ; // initialisiere Gewichte
7        $D_1(i) = \frac{1}{2a}$ ; // falls  $l_i = +1$ 
8        $D_1(i) = \frac{1}{2b}$ ; // falls  $l_i = -1$ 

9   Für  $t = 1, \dots, T$ ; // Schleife für Trainingsrunden
10      Für jedes  $i \in \{1 \dots m\}$ ;
11          setze  $D_t(i) = \frac{D_1(i)}{\sum_{j=1}^m D_1(j)}$ ; // Normalisierung der Gewichtsvektoren
12      Trainiere einen Weak Learner  $h_j$  für jedes Merkmal  $f_j$ , verwende  $D_t$ ;
13      Für jedes  $h_j$ ;
14           $r_j = \sum_{i=1}^m D_t(i) l_i h_j(x_i)$  mit  $h_j \in [+1, -1]$ ;
15      Wähle einen Weak Learner  $h_j$ , der  $|r_j|$  maximiert;
16      Für jedes  $i \in \{1 \dots m\}$ ; // Update der Gewichte
17           $D_{t+1}(i) = D_t(i) \exp(-\alpha_t l_i h_t(x_i))$ ;
18      wobei  $\alpha_t = \frac{1}{2} \ln\left(\frac{1+r_t}{1-r_t}\right)$ ;

```

**Rückgabe**

```

19    $H(x) = \text{sign}\left(\sum_{t=1}^T \alpha_t h_t(x)\right)$  // Strong Learener

```

---

**Abbildung C.2:** Der Discrete-Adaboost Trainingsalgorithmus



# Abbildungsverzeichnis

1.1	Zwei Roboter des Fachgebiets und deren Sensorik . . . . .	3
2.1	Beispiele für Personenhypothesen von drei unterschiedlichen Sensoren. . .	9
2.2	Laserscan mit lokalen Minima. . . . .	11
2.3	Occupancy maps für die Erkennung bewegter Objekte. . . . .	12
2.4	Merkmale für Beinsegmente. . . . .	13
2.5	Sensorik für Personendetektion in 3 Dimensionen. . . . .	15
2.6	Modelle zur Fusion von mehreren Sensorebenen. . . . .	16
2.7	Histogramme aus 3D Laserdaten. . . . .	17
2.8	Zweistufiges Detektionssystem von Spinello. . . . .	18
3.1	Erscheinungsformen von Beinen in Laserscans . . . . .	22
3.2	Die Verarbeitungsschritte einer Personendetektion. . . . .	24
3.4	Geometrische Relationen von kartesischen- und Polarkoordinaten. . . .	24
3.3	Schematische Darstellung eines 270° Laserscans . . . . .	25
3.5	Segmentierter Laserscan. . . . .	26
3.6	Verwendeter Algorithmus zum Segmentieren des Laserscans. . . . .	27
3.7	Schema des Adaboost Verfahrens. . . . .	32
3.8	Beispiel eines Decision Trees für die Beindetektion. . . . .	34
3.9	verwendeter Algorithmus zum generieren der Personenhypothese . . . .	36
3.10	Konzept der Crossvalidation. . . . .	39
3.11	Learning Curve eines Klassifikators. . . . .	40
4.1	Darstellung von Rollstühlen, Rollatoren und Personen mit Krücken . .	44

---

4.2	Schematische Darstellung für die Berechnung der GDIF Merkmale . . .	46
4.3	Darstellung zur Berechnung des Zentrums der Box. . . . .	48
4.4	Darstellung für die Berechnung der Bingrenzen $G_0, \dots, G_N$ . . . . .	48
4.5	Trigonometrische Relationen von Messpunkt und Mittellinie der Box für die Berechnung der lokalen Tiefenwerte. . . . .	49
5.1	Beispiele für die Negativdatensätze . . . . .	55
5.2	Evaluierungsgraphen für die Klassifikation . . . . .	59
5.3	Testergebnisse für die Segmentklassifikation mit 10 weak learnern und einer Baumtiefe von 1. . . . .	61
5.4	Testergebnisse für die Segmentklassifikation mit 50 weak learnern und einer Baumtiefe von 10. . . . .	61
5.5	Vergleich des BER auf allen Testumgebungen. . . . .	62
5.6	Testergebnisse für die Personenhypothese. . . . .	64
5.7	Evaluierungsergebnisse auf dem Krankenhausdatensatz . . . . .	65
5.8	Learning Curves mit unterschiedlicher maximaler Baumtiefe der weak learner und GDIF Merkmalen. . . . .	68
5.9	Evaluierungsergebnisse für Rollstühle und Rollatoren. . . . .	71
6.1	Vergleich vom Stand vor der Masterarbeit und jetzt. . . . .	74
A.1	Vergleich unterschiedlicher Adaboost Varianten für die Beindetektion. .	78
A.2	Vergleich der Beiden Methoden zur generierung der Personenhypothese.	79
B.1	alternative Methode zur Berechnung der lokalen Tiefenwerte . . . . .	82
B.2	Vergleich der durchschnittliche Laufzeiten der Merkmalsberechnung pro Segment . . . . .	83
C.1	Verwendeter algorithmus zum labeln der Trainingsdaten . . . . .	86
C.2	Der Discrete-Adaboost Trainingsalgorithmus . . . . .	87

---

# Literaturverzeichnis

- [ALEXANDER CARBALLO und YUTA, 2008] ALEXANDER CARBALLO, AKIHISA OHYA und S. YUTA (2008). *Fusion of Double Layered Multiple Laser Range Finders for People Detection from a Mobile Robot*. Multisensor Fusion and Integration for Intelligent Systems, 2008. MFI 2008. IEEE International Conference on, S. 677–682.
- [ARRAS et al., 2007] ARRAS, KAI O., ÓSCAR MARTÍNEZ MOZOS und W. BURGARD (2007). *Using Boosted Features for the Detection of People in 2D Range Data..* In: *ICRA*, S. 3402–3407. IEEE.
- [BENNEWITZ et al., 2005] BENNEWITZ, MAREN, W. BURGARD, G. CIELNIAK und S. THRUN (2005). *Learning motion patterns of people for compliant robot motion*. International Journal of Robotics Research, 24:31–48.
- [CAMPANA, 2011] CAMPANA, RICCARDO (2011). *People Detection and Tracking with Kinect for Mobile Platforms*. Diplomarbeit, Università di Padova Facoltà di Ingegneria.
- [CUI et al., 2005] CUI, JINSHI, H. ZHA, H. ZHAO und R. SHIBASAKI (2005). *Tracking multiple people using laser and vision..* In: *IROS*, S. 2116–2121. IEEE.
- [FERREIRA, 2007] FERREIRA, ARTUR (2007). *Survey on Boosting Algorithms for Supervised and Semi-supervised Learning*. Technischer Bericht, Instituto Superior de Engenharia de Lisboa.
- [FOD et al., 2002] FOD, AJO, A. HOWARD und M. J. MATARIC (2002). *Laser-based people tracking*. In: *In Proc. of the IEEE International Conference on Robotics & Automation (ICRA)*, S. 3024–3029.
-



- [FREUND und SCHAPIRE, 1995] FREUND, YOAV und R. E. SCHAPIRE (1995). *A Decision-Theoretic Generalization of on-Line Learning and an Application to Boosting*.
- [FRITSCH et al., 2004] FRITSCH, J., M. KLEINEHAGENBROCK, S. LANG, G. FINK und G. SAGERER (2004). *Audiovisual Person Tracking with a Mobile Robot*. In Proc. Int. Conf. on Intelligent Autonomous Systems, S. 898–906.
- [IKEMURA und FUJIYOSHI, 2011] IKEMURA, SHO und H. FUJIYOSHI (2011). *Real-time Human Detection Using Relational Depth Similarity Features*. In: *Proceedings of the 10th Asian Conference on Computer Vision - Volume Part IV*, ACCV’10, S. 25–38, Berlin, Heidelberg. Springer-Verlag.
- [KIDONO et al., 2011] KIDONO, KIYOSUMI, T. MIYASAKA, A. WATANABE, T. NAITO und J. MIURA (2011). *Pedestrian recognition using high-definition LIDAR..* In: *Intelligent Vehicles Symposium*, S. 405–410. IEEE.
- [KLEINEHAGENBROCK et al., 2002] KLEINEHAGENBROCK, M., S. LANG, J. FRITSCH, F. LOMKER, G. FINK und G. SAGERER (2002). *Person tracking with a mobile robot based on multi-modal anchoring*. In: *Robot and Human Interactive Communication, 2002. Proceedings. 11th IEEE International Workshop on*, S. 423–429.
- [KONDAXAKIS et al., 2009] KONDAXAKIS, POLYCHRONIS, H. BALTZAKIS und P. E. TRAHANIAS (2009). *Learning moving objects in a multi-target tracking scenario for mobile robots that use laser range measurements..* In: *IROS*, S. 1667–1672. IEEE.
- [LONG und SERVEDIO] LONG, PHILIP M. und R. A. SERVEDIO. *Random Classification Noise Defeats All Convex Potential Boosters*.
- [MARTIN et al., 2006] MARTIN, CHRISTIAN, E. SCHAFFERNICHT, A. SCHEIDIG und H.-M. GROSS (2006). *Multi-modal sensor fusion using a probabilistic aggregation scheme for people detection and tracking*. *Robotics and Autonomous Systems*, 54(9):721–728.
-

- 
- [MOZOS et al., 2009] MOZOS, OSCAR MARTINEZ, R. KURAZUME und T. HASEGAWA (2009). *Multi-Layer People Detection Using 2D Range Data*. In: *Proceedings of the ICRA 2009 Workshop: People Detection and Tracking*, Kobe, Japan.
- [NAVARRO-SERMENT et al., 2010] NAVARRO-SERMENT, LUIS ERNESTO, C. MERTZ und M. HEBERT (2010). *Pedestrian Detection and Tracking Using Three-dimensional LADAR Data*. The International Journal of Robotics Research, Special Issue on the Seventh International Conference on Field and Service Robots, 29(12):1516 – 1528.
- [NUNES und PREMEBIDA, 2005] NUNES, URBANO und C. PREMEBIDA (2005). *Segmentation and geometric primitives extraction from 2d laser range data for mobile robot applications*. In: *Robotica 2005 - Scientific meeting of the 5th National Robotics Festival*.
- [PREMEBIDA et al., 2009] PREMEBIDA, CRISTIANO, O. LUDWIG und U. NUNES (2009). *Exploiting LIDAR-based features on pedestrian detection in urban scenarios..* In: *Intelligent Transportation Systems, 2009. ITSC '09. 12th International IEEE Conference on*, S. 1–6.
- [SCHULZ et al., 2003] SCHULZ, DIRK, W. BURGARD, D. FOX und A. B. CREMERS (2003). *People Tracking with a Mobile Robot Using Sample-Based Joint Probabilistic Data Association Filters*.
- [SPINELLO et al., 2011] SPINELLO, L., M. LUBER und K. O. ARRAS (2011). *Tracking People in 3D Using a Bottom-Up Top-Down Detector..* In: *Proc. of The International Conference in Robotics and Automation (ICRA)*.
- [SPINELLO et al., 2010] SPINELLO, LUCIANO, K. O. ARRAS, R. TRIEBEL und R. SIEGWART (2010). *A Layered Approach to People Detection in 3D Range Data..* In: FOX, MARIA und D. POOLE, Hrsg.: *AAAI*. AAAI Press.
- [SPINELLO und SIEGWART, 2008] SPINELLO, LUCIANO und R. SIEGWART (2008). *Human detection using multimodal and multidimensional features..* In: *ICRA*, S. 3264–3269. IEEE.
-

- [TREPTOW et al., 2006] TREPTOW, ANDRÉ, G. CIELNIAK und T. DUCKETT (2006). *Real-time people tracking for mobile robots using thermal vision..* Robotics and Autonomous Systems, 54(9):729–739.
- [VIOLA und JONES, 2001] VIOLA, PAUL und M. JONES (2001). *Rapid object detection using a boosted cascade of simple features.* In: *Computer Vision and Pattern Recognition, 2001. CVPR 2001. IEEE Conference on*, Bd. 1, S. I–511–I–518 vol.1.
- [XAVIER et al., 2005] XAVIER, JOAO, M. PACHECO, D. CASTRO und A. RUANO (2005). *Fast line, arc/circle and leg detection from laser scan data in a player driver.* In: *in Proc. of the IEEE Int. Conference on Robotics & Automation (ICRA05)*.
- [ZHOU und HOANG, 2005] ZHOU, JIANPENG und J. HOANG (2005). *Real Time Robust Human Detection and Tracking System.* In: *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Workshops - Volume 03*, CVPR '05, S. 149–, Washington, DC, USA. IEEE Computer Society.
- [ZIVKOVIC und KROSE, 2007] ZIVKOVIC, Z. und B. KROSE (2007). *Part Based People Detection using 2D Range Data and Images.* In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Diego, USA.
-

# Thesen

- Detektion von Personen ist eine der Schlüsseleigenschaft für eine natürliche Mensch-Maschine-Interaktion.
- Der gegenwärtig am Fachgebiet eingesetzte Personendetektor mittels Klassifikation von Lasermessungen besitzt noch Verbesserungspotential hinsichtlich seiner Detektionsgenauigkeit.
- Des weiteren besteht die Einschränkung das nur unverdeckte Beine erkannt werden können. Dies ist aber in gewissen Umgebungen wie Krankenhäusern nicht immer gegeben, da hier Beine oft von Laufhilfen wie Krücken oder Rollatoren verdeckt werden können oder eine Person in einem Rollstuhl sitzt.
- Das vorgestellte Verfahren überwindet diese Einschränkungen, indem es nicht spezielle auf ein bestimmtes Objekt zugeschnittene geometrische Merkmale verwendet. Sondern direkt die Entfernungsmessungen des Lasersensors klassifiziert.
- Es wurde experimentell nachgewiesen, dass die neuen Histogramm basierten Merkmale generell (und teilweise deutlich) bessere Ergebnisse erzielen als der State of the Art Ansatz. Außerdem sind die Personenhypothesen nicht mehr nur auf Beine beschränkt, sondern erlauben ein weitaus variableres Aussehen von Personen.

Ilmenau, 21.02.2014

.....

Tim Wengefeld